Synthesis
Quick Reference

Version C-2009.06, June 2009

SYNOPSYS®

Copyright Notice and Proprietary Information

Contents

Getting Help

The synthesis tools provide various forms of online help.

- The help command provides you with quick help for one or more commands or procedures.
- The man command displays the man page.

You can use a wildcard pattern as the argument for the help command. The wildcard characters are

*	Matches n characters.
?	Matches exactly one character.

Accessing Brief Help

Use this command to list all commands by function group:

```
dc_shell> help
```

Use this command to display all commands that end with the word clock:

```
dc_shell> help *clock
```

Use this command to get syntax help for one or more commands:

```
dc_shell> help -verbose cmd_name_pattern
```

Use this command to get syntax help for a specific command:

```
dc_shell> command_name -help
```

Man Page Viewing Instructions

The following sections describe how to set up your environment and the syntax to use to view man pages.

Setting Up the UNIX Environment

Edit your .cshrc file to contain these lines:

```
setenv SYN_MAN_DIR synopsys_root/doc/syn/man
setenv MANPATH ${MANPATH}:${SYN_MAN_DIR}
```

SYN_MAN_DIR is a variable that contains the path to the man page directories, and *synopsys_root* represents the specific path to the Synopsys software directory at your site.

Viewing Man Pages From UNIX

Command

```
% man command_name
```

Variable

```
% man variable_name
```

Error, warning, or information message

```
% man message_id
```

Viewing Man Pages From dc_shell

Command

```
dc_shell> man command_name
```

Variable

```
dc_shell> man variable_name
```

Error, warning, or information message

```
dc_shell> man message_id
```

User Commands

Invoke user commands from a UNIX shell.

aman

Displays Synopsys extended error messages.

```
aman [error_message_code]
```

cache_ls

Lists elements in a Synopsys cache.

```
cache_ls cache_dir reg_expr
```

cache_rm

Removes elements from a Synopsys cache.

```
cache_rm cache_dir reg_expr
```

create_types

Extracts user-defined type information from VHDL package files.

```
create_types [-nc] [-w lib] [-v]
[-o logfile] file_list
```

dc_shell

Invokes the Design Compiler command shell.

```
dc_shell
[-f script_file]
[-x command_string]
[-no_init]
[-no_home_init]
[-no_local_init]
[-checkout feature_list]
[-64bit]
[-wait wait_time]
[-timeout timeout_value]
[-version]
[-no_log]
```

design_vision

Runs Design Vision visualization for Synopsys synthesis products.

```
design_vision [-f script_file]
[-x command_string]
[-no_init] [-checkout feature_list]
[-timeout timeout_value] [-version]
[-behavioral]
[-syntax_check | -context_check]
[-tcl_mode]
```

lc_shell

Runs the Library Compiler command shell.

```
[-f script_file]
[-x command_string]
[-no_init]
[-version]
```

synenc

Runs the Synopsys Encryptor for HDL source code.

```
synenc [-r synopsys_root] file_list
```

synopsys_users

Lists the current users of the Synopsys licensed features.

```
synopsys_users [feature_list]
```

Synthesis Commands

Invoke these commands from within the Design Compiler tool. Unless otherwise noted, all commands are available in both Design Compiler and Design Compiler topographical mode.

acs_check_directories

Checks the Automated Chip Synthesis (ACS) directory structure settings for correctness.

```
status acs_check_directories
[-verbose verbosity_level]
```

acs_compile_design

Compiles the constrained RTL to a netlist using constraints propagated from the top-level design.

```
status acs_compile_design
[-destination pass_name]
[-prepare_only]
[-force]
[-update]
[-update_source source_pass]
design_name
```

acs_create_directories

Creates the project directory tree for Automated Chip Synthesis.

```
int acs_create_directories
```

acs_customize_directory_structure

This command is obsolete. See the acs_user_dir variable for information about customizing the Automated Chip Synthesis directory structure.

acs_get_parent_partition

Creates a collection of designs that are compiled partitions containing the specified subdesign.

```
string acs_get_parent_partition
design_name
[-hierarchy]
[-list]
```

acs_get_path

Gets the path location for the specified file. To specify a file, you specify its file type and for pass-dependent files, its pass directory.

```
string acs_get_path
-file_type filetype
[-mode read | write]
[-pass pass_name]
[-name filename [-append]]
[-relative]
```

acs_merge_design

Preprocesses a design for incremental design update by merging the modified designs and their parent compile partitions with the mapped design being updated.

```
status acs_merge_design
[-auto_update | -update design_list]
[-unmapped source_dir]
[-mapped data_dir]
[-type pre | post]
[-destination dest_dir]
[-reference pre_compile_partition_dir]
design_name
```

acs_read_hdl

Reads in the HDL source code of a design and generates the GTECH representation.

```
status acs_read_hdl
[design_name]
[-hdl_source file_or_dir_list]
[-exclude_list file_or_dir_list]
[-format {verilog | vhdl}]
[-recurse]
[-no_dependency_check]
[-no_elaborate]
[-library design_lib_name]
[-verbose]
[-auto_update | -update file_list]
[-destination destination_dir]
[-sv_package_files file_list]
```

acs_recompile_design

Compiles an unmapped constrained .ddc file using time budgets. The time budgets are created by using a previously mapped design.

```
status acs_recompile_design
-budget_source budget_pass
-destination destination_pass
[-source source_pass]
[-prepare_only]
[-force]
[-update]
[-update_source source_pass]
design_name
```

acs_refine_design

Refines an already mapped design.

```
status acs_refine_design
[-source pass_name]
[-destination pass_name]
[-prepare_only]
[-force]
[-update]
[-update_source source_pass]
design_name
```

acs_remove_dont_touch

Removes the dont_touch attributes on the specified designs.

```
status acs_remove_dont_touch
[-force | design_list]
```

acs_report_attribute

Reports the ACS attributes on the specified partitions.

```
status acs_report_attribute
[-partitions partition_list]
[attribute_name]
```

acs_report_directories

Reports the current directory structure settings.

```
status acs_report_directories
[-file_types type_list]
```

acs_report_user_messages

Reports the number of error, warning, and information messages.

```
status acs_report_user_messages
[-total]
[-errors]
[-warnings]
[-infos]
[-reset]
```

acs_reset_directory_structure

Resets the project directory tree of Automated Chip Synthesis to its default setting.

```
status acs_reset_directory_structure
```

acs_set_attribute

Sets Automated Chip Synthesis compile attributes on one or more partitions.

```
status acs_set_attribute
[-partitions partitions]
attribute_name
[attribute_value]
```

acs_submit

Specifies the compile configuration for normal compile partitions.

```
status acs_submit
[-command command]
[-host host]
[-exec exec]
[-show]
```

acs_submit_large

Specifies the compile configuration for large compile partitions.

```
status acs_submit_large
[-command command]
[-host host]
[-exec exec]
[-partitions part_list]
[-show]
```

acs_write_html

Creates an HTML page in the destination directory.

```
status acs_write_html
-destination pass_name
[-acs_work_dir acs_working_directory]
```

add_module

Reads in a specified library file containing module functional information and uses it to update an existing technology library.

```
status add_module
[-overwrite]
[-force]
[-permanent]
file_name
library_name
[-no_warnings]
```

add_pg_pin_to_db

Converts rail or non-pg_pin based logic library (db) into pg_pin based logic library.

```
status add_pg_pin_to_db
input_db_filename
[-mw_library_name mw_lib_name]
[-pg_map_file pg.map]
-output pg_db_filename
[-verbose]
[-pg_map_template
pg_pin_map_template_filename]
[-expanded]
```

add_pg_pin_to_lib

converts rail or non-pg_pin based logic library (.lib) into pg_pin based logic library (.lib).

```
status add_pg_pin_to_lib
input_lib_filename
[-mw_library_name mw_lib_name]
[-pg_map_file pg.map]
[-pg_map_template template_filename]
[-expanded]
-output pg_lib_filename
[-common_shell_path common_shell_path]
[-verbose]
```

add_port_state

Adds state information to a supply port.

```
string add_port_state
supply_port_name
-state {name nom | min nom max | off}
```

add_pst_state

Defines the states of each of the supply nets for one possible state of the design.

```
status add_pst_state
state_name
-pst table_name
-state supply_states
```

add_to_collection

Adds objects to a collection, resulting in a new collection. The base collection remains unchanged.

```
collection add_to_collection
base_collection
object_spec
[-unique]
```

add_to_rp_group (Design Compiler topographical mode only)

Adds a cell, hierarchical group, or keepout to an existing relative placement group.

Leaf Cells:

```
status add_to_rp_group
rp_groups
-leaf cell_name
[-column integer]
[-row integer]
[-pin_align_name pin_name]
[-orientation direction]
[-alignment bottom-left | bottom-right]
```

Hierarchical Groups:

```
status add_to_rp_group
rp_groups
-hierarchy group_name
[-instance instance_name]
[-column integer]
[-row integer]
[-alignment bottom-left | bottom-right]
```

Keepouts:

```
status add_to_rp_group
rp_groups
-keepout keepout_name
[-column integer]
[-row integer]
[-width integer]
[-height integer]
[-type hard | soft | space]
```

alias

Creates a pseudo-command which expands to one or more words, or lists current alias definitions.

```
string alias
[name] [def]
```

alib_analyze_libs

Reads in the target library files, analyzes each of them separately, and creates the corresponding alib files.

```
int alib_analyze_libs
```

all_active_scenarios (Design Compiler topographical mode only)

Lists the active scenarios available in memory.

```
string all_active_scenarios
```

all_clock_gates

Returns a collection of clock gating cells or pins in the current design.

```
collection all_clock_gates
[-no_hierarchy]
[-clock clock_name]
[-cells]
[-enable_pins]
[-clock_pins]
[-output_pins]
[-test_pins]
[-observation_pins]
```

all_clocks

Returns a collection of all clocks in the current design.

```
collection all_clocks
```

all_connected

Returns the objects connected to a net, port, pin, net instance, or pin instance.

```
collection all_connected
[-leaf] object
```

all_critical_cells

Returns a collection of critical leaf cells in the top hierarchy of the current design.

```
collection all_critical_cells
[-slack_range range_value]
```

all_critical_pins

Returns a collection of critical endpoints or startpoints in the current design.

```
collection all_critical_pins
[-type endpoint | startpoint]
[-slack_range range_value]
```

all_designs

Returns a collection containing all designs in the current design.

```
collection all_designs
```

all_dont_touch

Returns a collection of dont_touch cells or nets from the current design or from the specified input collection.

```
collection all_dont_touch
-cells | -nets
[input_coll]
```

all_drc_violated_nets

Returns a collection of DRC-violated nets from the current design or from the specified input collection.

```
collection all_drc_violated_nets
-max_capacitance | -max_transition |
-max_fanout
[input_coll]
[-bound upper]
[-threshold threshold]
```

all_fanin

Reports pins, ports, or cells in the fanin of specified sinks.

```
collection all_fanin
-to sink_list
[-startpoints_only]
[-exclude_bboxes]
[-break_on_bboxes]
[-only_cells]
[-flat]
[-levels count]
```

all_fanout

Returns a set of pins, ports, or cells in the fanout of the specified sources.

```
collection all_fanout
-clock_tree
-from source_list
[-endpoints_only]
[-exclude_bboxes]
[-break_on_bboxes]
[-only_cells]
[-flat]
[-levels count]
```

all_high_fanout

Returns a collection of high-fanout nets from the current design or from the specified input collection.

```
collection all_high_fanout
-nets
[-threshold value]
[input_coll]
[-through_buf_inv]
```

all_ideal_nets

Returns a collection of ideal nets from the current design or from the specified input collection.

```
collection all_ideal_nets
[input_coll]
```

all_inputs

Returns a collection of input or inout ports in the current design.

```
collection all_inputs
[-clock clock_name]
[-edge_triggered | -level_sensitive]
```

all_operand_isolators

Returns a collection of operand isolation cells or pins in the current design.

```
collection all_operand_isolators
[-no_hierarchy]
[-cells]
[-control_pins]
[-data_pins]
[-output_pins]
```

all_outputs

Returns a collection of output or inout ports in the current design.

```
collection all_outputs
[-clock clock_name]
[-edge_triggered | -level_sensitive]
```

all_registers

Returns a collection of sequential cells or pins in the current design.

```
collection all_registers
[-no_hierarchy]
[-clock clock_name]
[-rise_clock rise_clock_name]
[-fall_clock fall_clock_name]
[-cells]
[-data_pins]
[-clock_pins]
[-slave_clock_pins]
[-output_pins]
[-inverted_output]
[-level_sensitive | -edge_triggered]
[-master_slave]
```

all_rp_groups (Design Compiler topographical mode only)

Returns a collection of specified relative placement groups and all groups in their hierarchy.

```
collection all_rp_groups
[rp_groups]
```

all_rp_hierarchicals (Design Compiler topographical mode only)

Returns a collection of hierarchical relative placement groups that contain specified groups in their hierarchy. The specified groups can be either included or instantiated in their parent group.

```
collection all_rp_hierarchicals
[rp_groups]
```

all_rp_inclusions (Design Compiler topographical mode only)

Returns a collection containing of hierarchical relative placement groups that include the specified groups.

```
collection all_rp_inclusions
[rp_groups]
```

all_rp_instantiations (Design Compiler topographical mode only)

Returns a collection of hierarchical relative placement groups that instantiate specified groups.

```
collection all_rp_instantiations
[rp_groups]
```

all_rp_references (Design Compiler topographical mode only)

Returns a collection of relative placement groups that directly contain the specified cells, which are either leaf cells or hierarchical cells that contain instantiated relative placement groups.

```
collection all_rp_references
[cell_list]
[-design design_name]
```

all_scenarios (Design Compiler topographical mode only)

Lists all defined scenarios available in memory.

```
string all_scenarios
```

all_threestate

Returns a collection of threestate cells or nets.

```
collection all_threestate
-nets
[input_coll]
```

all_tieoff_cells

Returns a collection of all tie-off cells in the current design or in the input collection.

```
collection all_tieoff_cells
[input_coll]
```

analyze

Analyzes the HDL files and stores the intermediate format in the specified library.

```
status analyze
[-library library_name]
[-work library_name]
[-format verilog | sverilog]
[-vcs vcs_opts]
[-create_update]
[-update]
[-define define_list]
file_list
```

append_to_collection

Adds objects to the collection in the specified variable. The variable is updated.

```
collection append_to_collection
var_name
object_spec
[-unique]
```

apply_clock_gate_latency

Annotates the clock latencies on the existing clock gating cells based on the settings previously specified using the set_clock_gate_latency command.

```
status apply_clock_gate_latency
```

apropos

Searches the command database for a pattern.

```
string apropos
[-symbols_only]
pattern
```

balance_buffer

Builds a balanced buffer tree on user-specified nets and drivers.

```
int balance_buffer
[-from start_point_list]
[-to end_point_list]
[-net net_list] [-force]
[-library library_name]
[-prefer buffer | inverter | lib_cell_name]
```

balance_registers

Moves the registers of the current design to achieve a minimum cycle time.

```
int balance_registers
```

break

Immediately exits a loop structure.

cd

Changes the current directory.

```
int cd
[directory]
```

cell_of

Returns the cell objects for the specified pins in the current design.

```
list cell_of
[object_list]
```

change_link

Changes the design to which a cell is linked.

```
status change_link
object_list
design_name
[-all_instances]
[-force]
```

change_macro_view (Design Compiler topographical mode only)

Changes the view of the macro that is used.

```
status change_macro_view
-reference cell_reference_name
-view view_name
[-quiet]
```

change_names

Changes the names of ports, cells, and nets in a design.

```
integer change_names
[-rules name_rules]
[-hierarchy]
[-verbose]
[-names_file names_file]
[-log_changes log_file]
[-restore]
[-dont_touch object_list]
[-instance instance]
[-new_name new_name]
```

change_selection

Changes the selection in the GUI, taking a collection of objects and changing the selection according to the type of change specified.

```
int change_selection
[-name slct_bus]
[-replace]
[-add]
[-remove]
[-toggle]
[-type object_type]
[-clock_trees clock_tree_list]
collection
```

change_selection_no_core

Indicates that objects had no core representation

```
change_selection_no_core -name Slct
[-add]
[-remove]
-type Type
-names NameList
```

change_selection_too_many_objects

Indicates that too many objects were involved in selection change

```
change_selection_too_many_objects -name Slct
[-add]
[-remove]
```

characterize

Captures information about the environment of specific cell instances, and assigns the information as attributes on the design to which the cells are linked.

```
int characterize
cell_list
[-no_timing] [-constraints]
[-connections] [-power]
[-verbose]
```

check_bindings

Checks the bindings in a synthetic library module definition.

```
int check_bindings
[-bindings binding_list]
[-pin_widths pin_width_list]
module_name
```

check_bsd

Checks whether a design's boundary-scan implementation is compliant with IEEE Std 1149.1.

```
int check_bsd
[-verbose true | false]
[-effort low | medium | high]
[-infer_instructions true | false]
```

check_budget

Checks that user-specified budgets and fixed delays are consistent with path constraints.

```
status check_budget
[-verbose]
[-tolerance tolerance]
[-from object_list]
[-to object_list]
[-no_interblock_logic]
[cell_list]
```

check_design

Checks the current design for consistency.

```
status check_design
[-summary]
[-no_warnings]
[-one_level]
[-multiple_designs]
[-no_connection_class]
[-post_layout | -only_post_layout]
```

check_error

Prints extended information on errors from last command.

```
int check_error
[-verbose] [-reset]
```

check_implementations

Checks the implementations in a synthetic library module definition.

```
int check_implementations
[-implementations implementation_list]
[-parameters parameter_list]
module_name
```

check_isolation_cells

Reports the existing isolation cells in the current design. It also reports if any isolation cell is redundant or might be required.

```
status check_isolation_cells
[-input]
[-output]
[-inside]
[-outside]
[objects]
```

check_level_shifters

Checks the design for all existing level shifters and nets against the specified level shifter strategy and threshold.

```
status check_level_shifters
[-verbose]
```

check_license

Checks the availability of a license for a feature.

```
status check_license
feature_list
```

check_mv_design

Checks for violations in a multivoltage design.

```
status check_mv_design
[-verbose]
[-isolation]
[-target_library_subset]
[-opcond_mismatches]
[-connection_rules]
[-level_shifters]
[-power_nets]
[-max_messages message_count]
```

check_noise

Checks whether there are necessary data available to run noise analysis in the current design.

```
status check_noise
[-verbose]
[-nosplit]
[-include check_list]
```

check_rp_groups (Design Compiler topographical mode only)

Checks the relative placement constraints and reports the failures.

```
collection check_rp_groups
{rp_groups | -all}
[-output filename]
```

check_scan_def

Allows scan chain structural consistency checking based on the scan chain information either stored in the current .ddc as an attribute or an ASCII SCANDEF file.

```
status check_scan_def
[-file file_name]
```

check_synlib

Performs semantic checks on synthetic libraries.

```
int check_synlib
```

check_target_library_subset

Checks and prints out the inconsistent settings among target library, target library subset, and operating conditions.

```
status check_target_library_subset
```

check_timing

Checks for possible timing problems in the current design.

```
status check_timing
[-overlap_tolerance minimum_distance]
[-override_defaults check_list]
[-multiple_clock]
[-retain]
[-include check_list]
[-exclude check_list]
```

check_tlu_plus_files (Design Compiler topographical mode only)

Checks the files used for TLUPlus extraction.

```
status check_tlu_plus_files
```

clean_buffer_tree

Removes the buffer tree at the specified driver pin on a mapped design.

```
int clean_buffer_tree
[-from start_point_list]
[-to end_point_list]
[-net net_list]
[-hierarchy]
[-global]
[-threshold integer_in_1]
```

close_mw_lib

Closes the current Milkyway library.

```
status close_mw_lib
```

compare_collections

Compares the contents of two collections. If the same objects are in both collections, the result is 0 (zero), like string compare. If they are different, the result is nonzero. The order of the objects can optionally be considered.

```
int compare_collections
[-order_dependent]
collection1
collection2
```

compare_delay_calculation

Compares the Arnoldi-based delays with the Elmore delays in the current design.

```
integer compare_delay_calculation
[-verbose]
[-ccs]
```

compare_interface_timing

Compares two write_interface_timing reports.

```
int compare_interface_timing
ref_timing_file
cmp_timing_file
[-output file_name]
[-absolute_tolerance atol_list]
[-nosplit]
[-significant_digit digits]
```

compare_lib

Performs a cross-reference check between a technology library and a symbol library or between a technology library and a physical library.

```
int compare_lib
library1
library2
```

compile

Performs logic-level and gate-level synthesis and optimization on the current design.

```
status compile
[-no_map]
[-map_effort medium | high]
[-area_effort none | low | medium | high]
[-incremental_mapping]
[-exact_map]
[-ungroup_all]
[-boundary_optimization]
[-auto_ungroup area | delay]
[-no_design_rule | -only_design_rule |
-only_hold_time]
[-scan]
[-top]
[-power_effort none | low | medium | high]
[-gate_clock]
```

compile_partitions

Distributes compile jobs for a design.

```
status compile_partitions
-destination pass
```

compile_ultra

Performs a high-effort compile on the current design for better quality of results (QoR).

```
status compile_ultra
[-incremental]
[-scan]
[-exact_map]
[-no_autoungroup]
[-no_seq_output_inversion]
[-no_boundary_optimization]
[-no_design_rule | -only_design_rule]
[-timing_high_effort_script |
-area_high_effort_script]
[-top]
[-retime]
[-gate_clock]
[-check_only]
[-num_cpus n]
[-congestion]
```

connect_logic_one

Connects the logic one net to a load pin on a cell instance.

```
status connect_logic_one
[-net_name net_name]
[load_pin_list]
[-reconnect_all]
[-mode power_intention | power_connection]
[-empty]
```

connect_logic_zero

Connects the logic zero net to a load pin on a cell instance.

```
int connect_logic_zero
[-net_name net_name]
[load_pin_list]
[-reconnect_all]
[-empty]
[-mode power_intention | power_connection]
```

connect_net

Connects the specified net to the specified pins or ports.

```
status connect_net
net object_list
```

connect_pin

Connects pins or ports at any level of hierarchy.

```
int connect_pin
-from from_object
-to to_list
-port_name port_name
-verbose
```

connect_power_domain

Connects power net information for the specified power domains. This command is supported only in non-UPF mode.

```
status connect_power_domain
power_domains
[-primary_power_net power_net]
[-primary_ground_net ground_net]
[-backup_power_net power_net]
[-backup_ground_net ground_net]
[-internal_power_net internal_power_net]
[-internal_ground_net internal_ground_net]
```

connect_power_net_info

Connects the specified power net information to the specified power pins. This command is supported only in non-UPF mode.

```
status connect_power_net_info
object_list
-power_pin_name power_pin_name
-power_net_name power_net_name
```

connect_supply_net

Connects the supply net to the specified supply ports and pins. This command is supported only in UPF mode.

```
status connect_supply_net
supply_net_name
-ports list
```

context_check

Enables or disables the Syntax Checker's context_check mode which checks commands for context errors.

```
integer context_check
true | false
```

continue

Begins the next loop iteration.

copy_collection

Duplicates the contents of a collection, resulting in a new collection. The base collection remains unchanged.

```
collection copy_collection
collection1
```

copy_design

Copies a design to a new design, or copies a list of designs to a new file in dc_shell memory.

```
string copy_design
design_list
target_name
```

copy_mw_lib

Copies a Milkyway library to another location.

```
status copy_mw_lib
[-from mw_lib | -from_lib_id lib_id]
-to lib_name
```

cputime

Reports the CPU time in seconds.

```
int cputime
[-all]
[-verbose]
```

create_bounds (Design Compiler topographical mode only)

Creates a fixed move bound or floating group bound in the design.

```
int create_bounds
[-name bound_name]
[-coordinate {llx1 lly1 urx1 ury1 ...}]
[-dimension {width height}]
[-effort low | medium | high | ultra]
[-type soft | hard]
[-exclusive]
object_list
```

create_bsd_patterns

Generates a set of functional patterns for a boundary-scan design.

```
status create_bsd_patterns
-output test_program_name
[-effort low | medium | high]
[-type all | functional | dc_parametric |
tap_controller | reset | tdr | bsr | leakage
| ac_input_pulse | ac_input_train |
ac_output_pulse | ac_output_train]
```

create_bus

Creates a port bus or a net bus.

```
status create_bus
object_list
bus_name
[-type type_name]
[-sort]
[-no_sort]
[-start start_bit]
[-end end_bit]
```

create_cache

Populates the cache directories with instances of the requested synthetic modules.

```
int create_cache
-module module_list
[-implementation implementation_list]
[-parameters parameter_list]
[-operating_condition operating_condition]
[-wire_load list] [-report]
```

create_cell

Creates cells in the current design or its subdesigns.

```
int create_cell
cell_list
reference_name
[-logic 0 | 1]
[-only_physical]
```

create_clock

Creates a clock object and defines its waveform in the current design.

```
status create_clock
[-name clock_name]
[-add]
[source_objects]
[-period period_value]
[-waveform edge_list]
```

create_command_group

Creates a new command group.

```
string create_command_group
group_name
```

create_design

Creates a design in dc_shell memory.

```
status create_design
design_name
[file_name]
```

create_die_area (Design Compiler topographical mode only)

Specifies the die area, rectangular or polygonal. This command is supported only in Topographical mode.

```
int create_die_area
-coordinate {llx lly urx ury}
-polygon {{x_0 y_0} {x_1 y_1} ... {x_n y_n}}
```

create_generated_clock

Creates a generated clock object.

```
string create_generated_clock
[-name clock_name]
[-add]
source_objects
-source master_pin
[-master_clock clock]
[-divide_by divide_factor
 | -multiply_by multiply_factor]
[-duty_cycle percent]
[-invert]
[-preinvert]
[-edges edge_list]
[-edge_shift edge_shift_list]
[-combinational]
```

create_ilm

Creates an interface logic model (ILM) for the current design. After command execution, the design in memory is the interface logic model.

```
status create_ilm
[-identify_only]
[-extract_only]
[-ignore_ports port_list]
[-no_auto_ignore]
[-latch_level levels]
[-keep_macros]
[-keep_boundary_cells]
[-keep_full_clock_tree]
[-include_side_load boundary | all | none]
[-traverse_disabled_arcs]
[-compact none | output | all]
[-case_controlled_ports port_list]
[-must_connect_ports port_list]
[-include_all_logic]
[-verbose]
```

create_multibit

Creates a multibit component for the specified list of cells or cell instances in the current design.

```
status create_multibit
object_list
[-name multibit_name]
[-sort]
[-no_sort]
```

create_mw_lib

Creates a Milkyway library.

```
status create_mw_lib
[-technology technology_file_name]
[-plib plib_file_name]
[-hier_separator sep]
[-bus_naming_style style]
[-mw_reference_library lib_list]
[-reference_control_file rc_file_name]
[-open]
libName
```

create_net

Creates nets in the current design or its subdesign.

```
status create_net
 | | | net_list
```

create_net_shape (Design Compiler topographical mode only)

Creates a new net shape.

```
collection create_net_shape
[-type wire | path | rect | poly]
-origin point
 | -points list_of_points
 | -bbox rect
 | [-length real]
[-width real]
[-path_type square | round |
extend_half_width | octagon]
-layer layer
-net net_name
[-vertical]
[-route_type route_type]
[-datatype int]
[-quiet]
[-net_type string]
```

create_nominal_power_state

Creates a new power state according to the format of the current power state template.

```
status create_nominal_power_state
voltage_list
[-name power_state_name]
```

create_operating_conditions

Creates a new set of operating conditions in a library.

```
int create_operating_conditions
-name name -library library_name
-process process_value
-temperature temperature_value
-voltage voltage_value
[-tree_type tree_type]
[-calc_mode calc_mode]
[-rail_voltages rail_value_pairs]
```

create_pass_directories

Creates the directory structure required for storing Automated Chip Synthesis data.

```
int create_pass_directories
pass_list
```

create_placement_blockage (Design Compiler topographical mode only)

Creates a new placement blockage.

```
status create_placement_blockage
-bbox rectangle
[-type hard | soft | partial]
[-blocked_percentage percentage]
[-name blockage_name]
```

create_port

Creates ports in the current design or its subdesign.

```
status create_port
port_list
[-direction dir]
```

create_power_domain

Creates a power domain, which provides a power supply distribution network.

create_power_net_info

Creates a power net.

```
status create_power_net_info
power_net_name
-power | -gnd
[-switchable]
[-nominal_voltages nominal_voltage_list]
[-voltage_ranges voltage_range_list]
```

create_power_switch

Creates a power switch at the specified power domain. This command is supported only in UPF mode.

```
string create_power_switch
switch_name
-domain domain_name
-output_supply_port {port_name
supply_net_name}
-input_supply_port {port_name
supply_net_name}
-control_port {port_name net_name}
[-ack_port {port_name net_name
[{boolean_function}]}]
[-ack_delay {port_name delay}]
-on_state {state_name input_supply_port
{boolean_function}}
[-off_state {state_name {boolean_function}}]
```

create_pst

Creates a power state table (PST), using a specific order of supply nets.

```
string create_pst
table_name
-supplies list
```

create_rp_group (Design Compiler topographical mode only)

Creates relative placement groups.

```
collection create_rp_group
group_list
[-design design_name]
[-columns num_cols]
[-rows num_rows]
[-alignment bottom-left | bottom-pin |
bottom-right]
[-pin_align_name pin_name]
[-utilization percentage]
[-ignore]
[-x_offset float]
[-y_offset float]
[-compress]
```

create_scenario (Design Compiler topographical mode only)

Creates a scenario in memory.

```
status create_scenario
scenario_name
```

create_site_row (Design Compiler topographical mode only)

Creates a row of sites.

```
collection create_site_row
-coordinate {X Y}
[-name row_name]
-kind site_type
-space space
-count site_count
[-orient orientation]
[-dir direction]
```

create_supply_net

Creates a supply net for the specified power domain. The supply net is created in the logic hierarchy at the same scope as the specified power domain. This command is supported only in UPF mode.

```
string create_supply_net
supply_net_name
-domain domain_name
[-reuse]
[-resolve unresolved | parallel]
```

create_supply_port

Creates a supply port in the specified power domain or in the current scope if no power domain is specified. This command is supported only in UPF mode.

```
string create_supply_port
supply_port_name
[-domain domain_name]
[-direction in | out]
```

create_test_protocol

Creates a test protocol based on user specification.

```
int create_test_protocol
[-infer_asynch]
[-infer_clock]
[-capture_procedure single_clock |
multi_clock]
```

create_voltage_area (Design Compiler topographical mode only)

Creates a voltage area at the specified region for providing placement constraints of cells associated with the region. This command is supported only in topographical mode.

```
int create_voltage_area
modules -name voltage_area_name
 | -power_domain power_domain_name
-coordinate llx1_lly1_urx1_ury1_...
[-guard_band_x guard_band_width]
[-guard_band_y guard_band_width]
[-is_fixed]
[-target_utilization utilization]
[-color string]
[-cycle_color]
```

create_wiring_keepouts (Design Compiler topographical mode only)

Creates a wiring keepout.

```
status create_wiring_keepouts
-name name
-layer layer
-coordinate list_of_float_values
```

current_design

Sets the working design.

```
string current_design
[design]
```

current_design_name

Returns the current design name.

```
string current_design_name
```

current_dft_partition

Sets or gets the design partition for the current specification.

```
status current_dft_partition
[design_partition_label]
```

current_instance

Sets the working instance object and enables other commands to be used on a specific cell in the design hierarchy.

```
string current_instance
[instance]
```

current_mw_lib

Gets the current Milkyway library.

```
collection current_mw_lib
```

current_scenario (Design Compiler topographical mode only)

Sets the current scenario.

```
string current_scenario
[scenario_name]
```

current_test_mode

Sets or gets the working test mode for the current design.

```
int current_test_mode
[test_mode_label]
```

date

Returns a string containing the current date and time.

```
string date
```

dc_allocate_budgets

Allocates budgets to specified cells.

```
string dc_allocate_budgets
[cell_list]
[-write_script]
[-file_format_spec format_spec]
[-format dctcl | dcsh]
[-separator hierarchy_separator]
[-mode rtl | gate | mixed]
[-budget_design_ware]
[-levels budget_levels]
[-no_interblock_logic]
[-min_register_to_output minimum_budget]
[-min_input_to_output minimum_budget]
[-min_input_to_register minimum_budget]
```

```
string dc_allocate_budgets
[cell_list]
[-write_script]
[-file_format_spec format_spec]
[-format dctcl | dcsh]
[-separator hierarchy_separator]
[-mode rtl | gate | mixed]
[-budget_design_ware]
[-levels budget_levels]
[-no_interblock_logic]
```

define_design_lib

Maps a design library to a UNIX directory.

```
int define_design_lib
library_name
-path directory
```

define_dft_design

Characterizes a design as a DFT design to be used for DFT insertion.

```
status define_dft_design
-design_name design_name
[-type design_type]
[-interface access_list]
[-test_model model_path_name]
[-params param_list]
```

define_dft_partition

Defines a design DFT partition to be created during DFT synthesis.

```
status define_dft_partition
design_partition_label
[-include list_of_cells_or_references]
[-clocks list_of_clocks]
[-rising_edge_clocks list_of_clocks]
[-falling_edge_clocks list_of_clocks]
```

define_name_rules

Defines a set of name rules for designs.

```
status define_name_rules
name_rules
[-max_length length]
[-target_bus_naming_style bus_naming_style]
[-allowed allowed_chars]
[-restricted restricted_chars]
[-first_restricted first_chars]
[-last_restricted last_chars]
[-reserved_words reserves]
[-replacement_char char]
[-remove_chars]
[-equal_ports_nets]
[-inout_ports_equal_nets]
[-collapse_name_space]
[-case_insensitive]
[-special output_format]
[-prefix prefix_name]
[-map map_string]
[-type object_type] [-reset]
[-remove_internal_net_bus]
[-remove_port_bus]
[-check_bus_indexing]
[-check_bus_indexing_use_type_info]
[-rename_three_state_port_net]
[-check_internal_net_name]
[-remove_irregular_port_bus]
[-remove_irregular_net_bus]
[-flatten_multi_dimension_busses]
[-dont_change_bus_members]
[-dont_change_ports]
[-add_dummy_nets]
[-dummy_net_prefix dummy_nets_format]
[-dir_inout_as_in]
```

define_proc_attributes

Defines attributes of a Tcl procedure, including an information string for help, a command group, a set of argument descriptions for help, and so on. The command returns the empty string.

```
string define_proc_attributes
proc_name
[-info info_text]
[-define_args arg_defs]
[-command_group group_name]
[-hide_body]
[-hidden]
[-dont_abbrev]
[-permanent]
```

define_scaling_lib_group

Defines a scaling library group to support voltage and/or temperature scaling.

```
status define_scaling_lib_group
[-name name]
[lib_file_names]
```

define_test_mode

Defines a test mode to be created during DFT synthesis.

```
status define_test_mode
test_mode_label
[-usage wrp_if | wrp_of | wrp_safe |
scan_compression | bist | bist_controller |
bist_scan | scan]
[-encoding {port_name 0 | 1, ...}]
[-inherit parent_mode_name]
```

define_user_attribute

Defines a new user-defined attribute.

```
int define_user_attribute
-type data_type
-classes class_list
[-quiet]
attr_name
[-range_max max]
[-one_of values]
[-range_min min]
[-import]
```

delete_operating_conditions

Deletes a specific set of operating conditions from a library.

```
status delete_operating_conditions
-library library_name
-name op_cond_name
```

derive_constraints

Propagates design environment, constraints, and attribute settings from the top-level design to the specified subdesigns.

```
int derive_constraints
[-attributes_only]
[-verbose]
[-budget]
cell_list
```

dft_drc

Checks the current design against test design rules.

```
int dft_drc
[-pre_dft] [-verbose]
[-coverage_estimate] [-sample percentage]
```

disconnect_net

Disconnects a net from pins or ports.

```
status disconnect_net
net
object_list | -all
```

disconnect_power_net_info

Removes the exceptional power net information hookups with the power pin.

```
status disconnect_power_net_info
object_list
-power_pin_name power_pin_name
```

drive_of

Returns the drive resistance value of the specified library cell pin.

```
float drive_of
library_cell_pin
[-rise | -fall]
[-piece best | worst | average
average_value]
 | [-min]
```

echo

Echos arguments to standard output.

```
string echo
[-n] [argument...]
```

elaborate

Builds a design from the intermediate format of a Verilog module, a VHDL entity and architecture, or a VHDL configuration.

```
status elaborate
design_name
[-library library_name
 | -work library_name]
[-architecture arch_name]
[-parameters param_list]
[-file_parameters file_list]
[-update]
[-ref]
```

encrypt_lib

Encrypts a VHDL source library file.

```
int encrypt_lib
file_name
[-output encrypted_file]
[-format {vhdl|tf|mwlib}]
[-key string]
```

error_info

Prints extended information on errors from last command.

```
string error_info
```

exit

Terminates the application.

```
string exit
[exit_code]
```

extract_physical_constraints (Design Compiler topographical mode only)

Extracts physical constraints information from one or more Design Exchange Format (DEF) files.

This command is supported only in topographical mode.

```
status extract_physical_constraints
def_files
[-verbose]
[-no_incremental]
[-exact]
```

extract_rc (Design Compiler topographical mode only)

Executes 2.5D extraction for routes in a design.

```
status extract_rc
[-estimate]
```

filter

Returns a list of design objects that satisfy a conditional attribute expression.

```
list filter
object_list
expression
[-regexp]
[-nocase]
```

filter_collection

Filters a collection, resulting in a new collection. The base collection remains unchanged.

```
collection filter_collection
base_collection
expression
[-regexp [-nocase]]
```

find

Finds a design or library object.

```
list find
type
[name_list]
[-hierarchy]
[-flat]
```

foreach

Specifies the control structure for list traversal loop execution.

foreach_in_collection

Iterates over the elements of a collection.

```
string foreach_in_collection
itr_var
collections
body
```

get_alternative_lib_cells

Creates a collection of equivalent library cells from loaded libraries, for a given cell or library cell. This collection can be used to replace or resize a specified cell in the current design. You can assign these library cells to a variable or pass them into another command.

```
string get_alternative_lib_cells
[-quiet]
[-regexp [-nocase]]
[-exact]
[-filter expression]
[-library libraries]
pattern_or_objects
```

get_always_on_logic

Returns a collection of cells and nets on always-on paths in the design.

```
collection get_always_on_logic
[-cells]
[-nets]
[-all]
```

get_app_var

Gets the value of an application variable.

```
string get_app_var
[-default | -details | -list]
[-only_changed_vars]
var
```

get_attribute

Returns the value of an attribute on a list of design or library objects.

```
list get_attribute
object_list
attribute_name
[-bus]
[-quiet]
```

get_buffers

Creates a collection of buffer cells from the libraries loaded in memory.

```
collection get_buffers
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-inverter]
[-inverting_buffers]
[-library lib_spec]
patterns
```

get_cells

Creates a collection of cells from the current design, relative to the current instance.

```
collection get_cells
[-hierarchical]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-filter expression]
[patterns | -of_objects objects]
```

get_clocks

Creates a collection of clocks from the current design.

```
collection get_clocks
[-quiet]
[-regexp]
[-nocase]
[-filter expression]
patterns
```

get_clusters

Creates a collection of one or more clusters.

```
collection get_clusters
[-quiet]
[-regexp [-nocase]]
[-exact]
[-filter expression]
[-hier]
[-flat]
patterns | -of_objects objects
```

get_command_option_values

Queries current/default option values.

```
get_command_option_values
[-default | -current]
-command command_name
```

get_cts_scenario (Design Compiler topographical mode only)

Returns the name of the clock tree synthesis scenario or the empty string if a clock tree synthesis scenario is not defined.

```
string get_cts_scenario
```

get_design_lib_path

Returns the directory to which the specified library is mapped.

```
status get_design_lib_path
library_name
```

get_designs

Creates a collection of one or more designs loaded into the tool.

```
collection get_designs
[-hierarchical]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-filter expression]
patterns
```

get_generated_clocks

Creates a collection of generated clocks.

```
collection get_generated_clocks
[-quiet]
[-regexp]
[-nocase]
[-filter expression]
[-exact]
patterns
```

get_ilm_objects

Returns a collection of nets, cells, or pins that are part of the interface logic models for the current design.

```
collection get_ilm_objects
[-type net | pin | cell]
```

get_ilms

Creates a collection of interface logic models (ILMs) defined in the current design.

```
collection get_ilms
[-quiet]
[-reference]
[-filter expression]
patterns
[-nocase]
[-exact]
[-regexp]
```

get_lib_attribute

Returns the value of an attribute on a list of library objects.

```
list get_lib_attribute
object_list
attribute_name
```

get_lib_cells

Creates a collection of library cells from the libraries loaded into memory.

```
collection get_lib_cells
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-scenario scenario_name]
[patterns]
[-of_objects objects]
```

get_lib_pins

Creates a collection of library cell pins from libraries loaded into memory.

```
collection get_lib_pins
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[-exact]
patterns | -of_objects objects
```

get_libs

Creates a collection of libraries loaded into memory.

```
collection get_libs
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-scenario scenario_name]
[-of_objects objects]
[patterns]
```

get_license

Obtains a license for a feature.

```
status get_license
feature_list
```

get_message_info

Returns information about diagnostic messages.

```
Integer get_message_info
[-error_count | -warning_count | -info_count
|
-limit l_id
 | -occurrences o_id
 | -suppressed s_id]
```

get_multibits

Creates a collection of one or more multibits loaded into dc_shell. You can assign these multibits to a variable or pass them into another command.

```
string get_multibits
[-quiet]
[-regexp]
[-nocase]
[-filter expression]
[patterns]
```

get_nets

Creates a collection of nets that meet the specified criteria.

```
collection get_nets
[-hierarchical]
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-top_net_of_hierarchical_group]
[-segments]
patterns | -of_objects objects
```

get_object_name

Returns the name of the object in a single-object collection.

```
string get_object_name
collection
```

get_path_groups

Creates a collection of path groups from the current design.

```
collection get_path_groups
[-quiet]
[-regexp]
[-nocase]
[-filter expression]
patterns
```

get_physical_hierarchy (Design Compiler topographical mode only)

Returns a list of hierarchical cells that are physical hierarchies in the current design.

```
integer get_physical_hierarchy
```

get_pins

Creates a collection of pins that match the specified criteria.

```
collection get_pins
[-hierarchical]
[-filter expression]
[-quiet]
[-regexp [-nocase] | -exact]
[patterns | -of_objects objects [-leaf]]
```

get_ports

Creates a collection of ports from the current design.

```
collection get_ports
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-filter expression]
patterns | -of_objects objects
[-hierarchical]
```

get_power_domains

Creates a collection of power domains that meet the specified criteria.

UPF Mode:

```
collection get_power_domains
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[[patterns] [-hierarchical] | -of_objects
objects]
```

Non-UPF Mode:

```
collection get_power_domains
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[patterns]
[-of_objects objects]
```

get_power_switches

Creates a collection of power switches that meet the specified criteria. This command is supported only in UPF mode.

```
collection get_power_switches
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[patterns]
[-hierarchical]
```

get_references

Creates a collection of one or more references loaded into memory.

```
collection get_references
[-hierarchical]
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-filter expression]
patterns
```

get_rp_groups

Creates a collection of relative placement groups.

```
collection get_rp_groups
[-quiet]
[-regexp]
[-nocase]
[-exact]
[-ignored]
[-top]
[patterns | -of_objects objects]
```

get_scan_cells_of_chain

Returns a collection containing all scan cells of the specified scan chain.

```
collection get_scan_cells_of_chain
-chain chain_name
[-test_mode test_mode]
```

get_scan_chains

Returns the number of scan chains in the current design.

```
status get_scan_chains
[-test_mode test_mode]
```

get_selection

Returns a collection containing the current selection in the GUI.

```
collection get_selection
[-slct_targets target_selection_bus
[-slct_targets_operation operation]]
-create_slct_buses
[-name selection_bus]
[-type object_type]
[-design design]
[-more_than more]
[-fewer_than fewer]
[-count]
[-num num]
```

get_supply_nets

Creates a collection of supply nets that meet the specified criteria. This command is supported only in UPF mode.

```
collection get_supply_nets
[-filter expression]
[-quiet]
[-regexp]
[-nocase]
[patterns]
[-hierarchical]
```

get_supply_ports

Creates a collection of supply ports that meet the specified criteria. This command is supported only in UPF mode.

```
collection get_supply_ports
[-filter expression]
[-quiet]
[-regexp [-nocase]]
[patterns]
[-hierarchical]
```

get_timing_paths

Creates a collection of timing paths for custom reporting and other processing.

```
collection get_timing_paths
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-delay_type delay_type]
[-nworst paths_per_endpoint]
[-max_paths max_path_count]
[-enable_preset_clear_arcs]
[-group group_name]
[-true [-true_threshold path_delay]]
[-slack_greater_than greater_slack_limit]
[-slack_lesser_than lesser_slack_limit]
[-greater greater_limit]
[-lesser lesser_limit]
[-include_hierarchical_pins]
[-path_type full_clock_expanded | full]
```

get_unix_variable

This is a synonym for the getenv command.

get_zero_interconnect_delay_mode

Reports whether or not the timer is currently using zero interconnect delay mode.

```
status get_zero_interconnect_delay_mode
```

getenv

Returns the value of a system environment variable.

```
string getenv
variable_name
```

group

Creates a new level of hierarchy.

```
status group
[cell_list | -logic | -pla | -fsm]
[-soft
 | -hdl_block block_name
 | -hdl_all_blocks
 | -hdl_bussed]
[-design_name design_name]
[-cell_name cell_name]
[-except exclude_list]
```

group_path

Groups a set of paths for cost function calculations.

```
int group_path
[-weight weight_value]
[-critical_range range_value]
-default | -name group_name
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
```

group_variable

Adds a variable to the specified variable group. This command is typically used by the system administrator only.

```
int group_variable
group_name variable_name
```

gui_bin

Bin a collection

```
string gui_bin -clct Clct
[-attr Attr]
[-cmd Cmd]
[-lower_bound LBound]
[-lower_bound_strict]
[-upper_bound UBound]
[-upper_bound_strict]
[-boundary Boundary]
[-num_bins numBins]
[-bin_range inRange]
[-underflow]
[-overflow]
[-nice_level niceLevel]
[-small_is_good]
[-exact_binning]
[-ignore_values ignoreList]
[-bar_brush brushPattern]
[-create_slct_buses]
[-filter_cmd filterCmd]
[-numBin numBins]
[-return_values]
[-slct_targets slctTargets]
[-slct_targets_operation slctTargetOp]
[-value_list valList]
```

gui_change_highlight

Manipulate the set of globally highlighted objects.

```
string gui_change_highlight
[-add | -remove | -toggle]
[-color color_id | -all_colors]
[-collection clct]
```

gui_create_attrdef

Creates the user-defined attribute for the design objects.

```
status_value gui_create_attrdef
 -class design_object
 -name attribute_name
 -get_value_cmd get_value_tcl_command
 -clct_and_values
collection_and_values_list_pair
 [-set_value_cmd set_value_tcl_command]
 [-enum_list enum_list]
 [-model_kind model_name]
 [-type "int | double | boolean | string"]
 [-format format_string]
 [-display_name display_name]
 [-subtype "name | full_name"]
 [-width display_width]
 [-show | -hide]
 [-show_infotip | -hide_infotip]
 [-custom_edit_name custom_edit_name]
 [-custom_edit_args custom_edit_args]
 [-is_editable_cmd is_editable_tcl_command]
```

gui_create_attrgroup

Creates the group of attributes for the design object.

```
status_value gui_create_attrgroup
 -class design_object (name of design
object)
 -name name (attribute group name)
 [-model_kind model_name] (model kind, if
omitted - default model kind)
 [-attr_list {{attr_1}...{attr_n}}]
 (list of attributes)
```

gui_create_pref_category

Create a preference category.

```
string gui_create_pref_category -category
Category
```

gui_create_pref_key

Create a preference key.

```
string gui_create_pref_key -key Key
-value_type ValueType
-value Value
[-category Category]
```

gui_delete_attrdef

Deletes the user-defined attribute for the design objects.

```
status_value gui_delete_attrdef
 [-class design_object]
 [-model_kind model_name]
 [-name attribute_name]
 [-all]
```

gui_delete_attrgroup

Deletes the group of attributes for the design object.

```
status_value gui_delete_attrgroup
 -class design_object (name of design
object)
 -name name (attribute group name)
 [-model_kind model_name] (model kind, if
omitted - default model kind)
 [-all] (delete all groups)
```

gui_exist_pref_category

Check the existence of a preference category.

```
bool gui_exist_pref_category -category
Category
```

gui_exist_pref_key

Check the existence of a preference key.

```
bool gui_exist_pref_key -key Key
[-category Category]
```

gui_get_current_task

Get the name of the current task.

```
string gui_get_current_taskP
```

gui_get_highlight

Get a collection of highlighted objects.

```
string gui_get_highlight
[-color color_id | -all_colors]
```

gui_get_highlight_options

Query the options that control highlighting.

```
string gui_get_highlight_options
[-current_color | -all_colors |
-auto_cycle_color]
```

gui_get_pref_keys

Return a list of preference keys under the specified category.

```
string gui_get_pref_keys -category Category
```

gui_get_pref_value

Get the value of a preference key.

```
string gui_get_pref_value -key Key
[-category Category]
```

gui_get_setting

Gets a setting on the specified window.

```
string gui_get_setting
-window WindowID
-setting Setting
```

gui_get_current_task_list

List all available task names.

```
string gui_get_task_listP
```

gui_get_window_ids

Get a list of window ids

```
ids gui_get_window_ids
[ -parent window_id ]
[ -type window_type ]
```

gui_get_window_pref_categories

Get list of preference categories for object specified by window

```
categories gui_get_window_pref_categories
{ -window window_id | -window_type
window_type }
```

gui_get_window_pref_keys

Get list of preference categories for object specified by window

```
categories gui_get_window_pref_keys
{ -window window_id | -window_type
window_type }
[ -category category ]
```

gui_get_window_pref_value

Get preference value for object specified by window or window type

```
value gui_get_window_pref_value
{ -window window_id | -window_type
window_type }
[ -category category ]
-key key
```

gui_get_window_types

Get a list of window types

```
types gui_get_window_types
[ -type token ]
```

gui_list_attrdefs

Lists user-defined attributes.

```
status_value gui_list_attrdefs
 [-class design_object]
 [-model_kind model_name]
 [-name attrdef_name]
 [-all]
 [-tcl]
 [-full]
```

gui_list_attrgroup

Lists the group of attributes for the design object.

```
status_value gui_list_attrgroup
 -class design_object (name of design
object)
 -name name (attribute group name)
 [-model_kind model_name] (model kind, if
omitted - default model kind)
 [-all] (all object classes)
 [-tcl] (tcl output format)
 [-full] (full tcl output format)
 [-attr_list] (list attributes only)
```

gui_remove_pref_key

Remove a preference key.

```
string gui_remove_pref_key -key Key
[-category Category]
```

gui_set_current_task

Set current task to the given task.

```
string gui_set_current_task -name task_name
-name task_name
```

gui_set_highlight_options

Change the options that control highlighting.

```
string gui_set_highlight_options
[-current_color color_id | -next_color |
-auto_cycle_colors enable]
```

gui_set_pref_value

Set the value of a preference key.

```
string gui_set_pref_value -key Key -value
Value
[-category Category]
```

gui_set_setting

Sets a setting on the specified window.

```
gui_set_setting
-window WindowID
-setting Setting
-value Value
```

gui_set_window_pref_key

Create a preference key owned by a particular window or window type

```
new_value gui_set_window_pref_key
{ -window window_id | -window_type
window_type }
[ -category category ]
-key key
-value_type value_type
-value value
```

gui_show_man_page

Show a man page in the man browser

```
string gui_show_man_page topic [-apropos]
```

gui_start

Starts the application GUI.

```
string gui_start
[-file name_of_script_file]
[-no_windows]
[-- x_args ...]
```

gui_stop

Stops the application GUI.

```
string gui_stop
```

gui_update_attrdef

Updates the user-defined attribute for the design objects.

```
status_value gui_update_attrdef
 -class design_object
 -name attribute_name
 [-get_value_cmd get_value_tcl_command]
 [-set_value_cmd set_value_tcl_command]
 [-enum_list enum_list]
 [-model_kind model_name]
 [-type "int | double | boolean | string"]
 [-format format_string]
 [-display_name display_name]
 [-subtype "name | full_name"]
 [-width display_width]
 [-show | -hide]
 [-show_infotip | -hide_infotip]
 [-custom_edit_name custom_edit_name]
 [-custom_edit_args custom_edit_args]
```

gui_update_attrgroup

Updates the group of attributes for the design object.

```
status_value gui_update_attrgroup
 -class design_object (name of design
object)
 -name name (attribute group name)
 [-model_kind model_name] (model kind, if
omitted - default model kind)
 [-attr_list {{attr_1}...{attr_n}}]
 (list of attributes)
 [-add] (add new attribute to the group)
 [-delete] (delete attribute from the group)
 [-move destination] (move attribute in the
list:
 Values: up, down, top, bottom, after,
 before)
 [-attr attribute] (single attribute)
 [-anchor attribute] (anchor attribute)
```

gui_update_pref_file

Save current application preferences to the user preference file.

```
string gui_update_pref_file [-file
FilePathName]
```

help

Displays quick help for one or more commands.

```
string help
[-verbose] [-groups] [pattern]
```

history

Displays or modifies the commands recorded in the history list.

```
string history
[-h] [-r] [args...]
```

hookup_power_gating_ports

Connects the control pins of retention registers through the design hierarchy based on the specification.

```
status hookup_power_gating_ports
[-type style]
[-default_port_naming_style naming_style]
[-port_naming_styles
list_of_port_naming_styles]
```

hookup_retention_register

Hooks up the save and restore pins of the retention registers to the save signal and the restore signal. This command is supported only in UPF mode.

```
status hookup_retention_register
```

identify_clock_gating

Identifies Power Compiler inserted clock-gating circuitry in a structural netlist.

```
integer identify_clock_gating
[-reset]
[-reset_only cell_or_pin_list]
[-gating_element gating_cell]
[-gated_element gated_cell_or_pin_list]
[-ungated_element ungated_cell_list]
```

if

Conditional execution control structure.

index_collection

Given a collection and an index, if the index is in range, extracts the object at that index and creates a new collection containing only that object. The base collection remains unchanged.

```
collection index_collection
collection1
index
```

infer_power_domains

Infers power domains from the $power specification.

```
status infer_power_domains
[-verbose]
```

insert_buffer

Inserts buffer cells on nets connected to specified pins.

```
collection insert_buffer
[-new_net_names new_net_names]
[-new_cell_names new_cell_names]
[-no_of_cells number]
[-inverter_pair]
object_list
buffer_lib_cell
```

insert_clock_gating

Performs clock gating on an appropriately-prepared GTECH netlist.

```
status insert_clock_gating
[-regular_only]
[-global]
[-no_hier]
```

insert_dft

Adds scan circuitry (either internal scan or boundary scan) to the current design.

```
int insert_dft
```

insert_isolation_cell

Inserts isolation cells on the specified nets, pins or ports. Isolation cell is a general term that applies to isolation (ISO) cells and enabled level-shifter (ELS) cells.

```
status insert_isolation_cell
[-force]
[-verbose]
-enable enable_signal
-object_list objects
-reference lib_cell_name
```

insert_level_shifters

Inserts appropriate level shifters in the current design.

```
status insert_level_shifters
[-preserve]
[-all_clock_nets]
[-clock_net clock_name]
[-verbose]
```

is_false

Tests the value of a specified variable, and returns a 1 if the value is 0 or the case-insensitive string *false*; returns a 0 if the value is 1 or the case-insensitive string *true*.

```
int is_false
value
```

is_true

Tests the value of a specified variable, and returns a 1 if the value is 1 or the case-insensitive string *true*; returns a 0 if the value is 0 or the case-insensitive string *false*.

```
int is_true
value
```

lib2saif

Creates a forward-annotation SAIF file for a specified technology library.

```
int lib2saif
[-output file_name]
library
[-lib_pathname lib_path_name]
```

license_users

Lists the current users of the Synopsys licensed features.

```
status license_users
[feature_list]
```

link

Resolves design references.

```
int link
[-remove_sub_design]
```

list

Creates a list.

```
int list arg1 arg2 arg arg ...
```

list_attributes

Lists the currently defined attributes.

```
string list_attributes
[-application]
[-class class_name]
```

list_designs

List the designs available in memory.

```
int list_designs
[design_list]
[-show_file]
```

list_duplicate_designs

Lists designs that have the same design name in dc_shell.

```
int list_duplicate_designs
```

list_files

Lists the files that are loaded into memory.

```
integer list_files
```

list_instances

Lists the instances in the current design or current instance.

```
int list_instances
[instance_list]
[-hierarchy]
[-max_levels num_levels]
[-full]
```

list_libs

Lists the libraries available in memory.

```
status list_libs
[lib_list]
```

list_licenses

Displays a list of licenses currently checked out by the user.

```
status list_licenses
```

list_test_models

Lists the designs with test models available in dc_shell.

```
status list_test_models
```

list_test_modes

Displays all of the test modes that are defined for the current design.

```
int list_test_modes
```

lminus

Removes one or more named elements from a list and returns a new list.

```
list lminus
[-exact] the_list elements
```

load_of

Returns the capacitance of the specified library cell pin.

```
float load_of
library_cell_pin
```

load_upf

Reads a script in the Unified Power Format (UPF). This command is supported only in UPF mode.

```
status load_upf
upf_file_name
[-scope instance_name]
[-noecho]
```

ls

Lists the contents of a directory.

```
string ls [filename ...]
```

man

Displays reference manual pages.

```
string man
topic
```

map_isolation_cell

Specifies how to map or remap the isolations and enable level-shifter cells belonging to the specified isolation strategy. This command is supported only in UPF mode.

```
status map_isolation_cell
isolation_strategy
-domain power_domain
-lib_cells lib_cells
```

map_level_shifter_cell

Specifies that the level-shifter cells belonging to the specified strategy can only be mapped to a subset of the library cells. This command is supported only in UPF mode.

```
status map_level_shifter_cell
level_shifter_strategy
-domain power_domain
-lib_cells lib_cells
```

map_power_switch

Defines which power switch library cells to use for the mapping of the given UPF power switch.

```
status map_power_switch
switch_name
-domain domain_name
-lib_cells name_list
```

map_retention_cell

Defines how to map the unmapped sequential cells to retention cells for the specified UPF retention strategy of the power domain. This command is supported only in UPF mode.

```
status map_retention_cell
retention_strategy
-domain power_domain
[-lib_cells lib_cells]
[-lib_cell_type lib_cell_type]
[-elements objects]
```

mem

Reports memory usage information.

```
int mem
[-all]
[-verbose]
```

merge_saif

Reads a list of SAIF files with their corresponding weights, computes the merged toggle rate and static probability, and annotates the switching activity for nets, pins,and ports in the current design. The command then generates a merged output SAIF file.

```
integer merge_saif
-input_list saif_file_and_weight_list
-instance_name inst_name
[-output merged_saif_name]
[-simple_merge]
[-ignore ignore_name]
[-ignore_absolute ig_absolute_name]
[-exclude exclude_file_name]
[-exclude_absolute ex_absolute_file_name]
[-unit_base unit_value]
[-scale scale_value]
[-khrate khrate_value]
[-map_names]
[-rtl_direct]
[-strip_module annotated_instance_name]
```

name_format

Specifies the name format for the isolation cells and level shifters.

```
status name_format
[-isolation_prefix name]
[-isolation_suffix name]
[-level_shift_prefix name]
[-level_shift_suffix name]
```

open_mw_lib

Opens a Milkyway library.

```
collection open_mw_lib
[-readonly | -write_ref]
mw_lib
```

optimize_registers

Performs retiming of sequential cells (edge-triggered registers or level-sensitive latches) on a mapped gate-level netlist; determines the placement of sequential cells in a design to achieve a target clock period; and minimizes the number of sequential cells while maintaining that clock period.

```
int optimize_registers
[-minimum_period_only]
[-no_compile]
[-sync_transform multiclass | decompose |
dont_retime]
[-async_transform multiclass | decompose |
dont_retime]
[-check_design [-verbose]]
[-print_critical_loop]
[-clock clock_name [-edge rise | fall]]
[-latch]
[-justification_effort low | medium | high]
[-only_attributed_designs]
```

parse_proc_arguments

Parses the arguments passed into a Tcl procedure.

```
string parse_proc_arguments -args arg_list
result_array
```

pipeline_design

Pipelines combinational designs by adding registers at the outputs and by retiming the circuit.

```
integer pipeline_design
[-stages number_of_stages]
[-stall_ports port_list]
[-stall_polarity high | low]
[-sync_reset reset_port
 | -async_reset reset_port]
[-reset_polarity high | low]
[-clock_port_name clock_port]
[-check_design [-verbose]]
[-print_critical_loop]
[-minimum_period_only]
[-register_outputs]
[-exact_map]
[-no_compile]
```

preview_dft

Previews, but does not implement, the test points, scan chains, and on-chip clocking control logic to be added to the current design.

```
integer preview_dft
[-show bidirectionals | cells | scan |
scan_clocks | scan_signals | segments |
tristates | voltages | power_domains |
scan_summary | all]
[-test_points all]
[-test_mode mode_name]
[-test_wrappers all]
[-bsd cells | data_registers | instructions
| tap | all]
[-script]
[-verbose]
```

print_message_info

Prints information about diagnostic messages which have occurred or have been limited.

```
string print_message_info
[-ids id_list] [-summary]
```

print_proc_new_vars

Check for new variables created within a Tcl procedure.

```
string print_proc_new_vars
```

print_suppressed_messages

Displays an alphabetical list of message ids that are currently suppressed.

```
string print_suppressed_messages
```

print_variable_group

Lists the variables defined in a specified variable group, along with their current values.

```
status print_variable_group
group
```

printenv

Prints the value of environment variables.

```
string printenv
[variable_name]
```

printvar

Prints the values of one or more variables.

```
string printvar
[pattern] [-user_defined] [-application]
```

proc_args

Displays the formal parameters of a procedure.

```
string proc_args
proc_name
```

proc_body

Displays the body of a procedure.

```
string proc_body
proc_name
```

propagate_annotated_delay_up

This command is obsolete and has been replaced by the propagate_ilm command. For more information, see the man page for the propagate_ilm command.

propagate_constraints

Propagates timing constraints from lower levels of the design hierarchy to the current design.

```
status propagate_constraints
[-design design_list]
[-all]
[-clocks]
[-disable_timing]
[-dont_apply]
[-false_path]
[-gate_clock]
[-ideal_network]
[-ignore_from_or_to_port_exceptions]
[-ignore_through_port_exceptions]
[-max_delay]
[-min_delay]
[-multicycle_path]
[-operating_conditions]
[-power_supply_data]
[-output file_name]
[-port_isolation]
[-verbose]
[-case_analysis]
[-target_library_subset]
```

propagate_placement

Propagates placement information for all leaf cells in the specified list of hierarchical cells from their respective reference designs to the current design.

```
int propagate_placement
[-verbose]
cell_list
```

propagate_switching_activity

Forces a propagation of the power switching activity information.

```
int propagate_switching_activity
[-effort low | medium | high]
[-verbose]
[-infer_related_clocks]
```

push_down_model

Used to create a new hierarchy around the current design for use with wrapping models flow in SoCTest.

```
int push_down_model
new_design_name
```

pwd

Displays the pathname of the present working directory (pwd), also called the current directory.

```
string pwd
```

query_objects

Searches for and displays objects in the database.

```
string query_objects
[-verbose]
[-class class_name]
[-truncate elem_count]
object_spec
```

quit

Exits the shell.

```
string quit
```

quit!

quits the application without posting an application exit dialog

```
quit!
```

read_bsdl

Reads the boundary-scan description language (BSDL) file for a boundary-scan design.

```
status read_bsdl
file_name
```

read_db

Reads in one or more design or library files in Synopsys database (.db) format.

```
string read_db
file_names
```

read_ddc

Reads in one or more design files in .ddc (Synopsys logical database) format.

```
status read_ddc
file_names
[-scenarios scenario_list]
[-active_scenarios active_scenario_list]
```

read_file

Reads designs or libraries into memory, or reads libraries into the shell.

```
list read_file
file_list
[-define macro_names]
[-format format_name]
[-ilm]
[-library library_name]
[-rtl]
[-single_file single_file_name]
[-work library_name]
[-scenarios scenario_list]
[-active_scenarios active_scenario_list]
[-names_file file_list]
```

read_lib

Reads a technology library, physical library, or symbol library into the shell. Creates a physical library for GDSII.

```
int read_lib
[-format format_name]
[-symbol intermediate_symbol_library_file]
[-plibrary physical_library_output_file]
[-pplibrary pseudo_physical_file_name]
file_name
[-no_warnings]
[-names_file file_list]
[-test_model CTL_file_list]
[-html]
[-lib_messages
{predefined_lib_message_variables}]
```

read_parasitics

Reads net parasitics information from an SPEF, DSPF, or RSPF file, and uses it to annotate the current design.

```
int read_parasitics
[-syntax_only]
[-elmore | -arnoldi]
[-increment] [-pin_cap_included]
[-net_cap_only] [-complete_with zero | wlm]
[-path path_name]
[-strip_path path_name]
[-quiet | -verbose]
[-dont_write_to_db]
[file_list]
```

read_partition

Reads the database for a design. For use only in dc_shell-t (Tcl mode of dc_shell).

```
int read_partition
-source pass
-type pre | post
design_name
[-format db | ddc]
```

read_pin_map

Reads in a port-to-pin mapping file, which defines the design port-to-package pin mapping for a boundary-scan design.

```
status read_pin_map
path_name
```

read_saif

Reads a SAIF file and annotates switching activity information on nets, pins, ports, and cells in the current design.

```
status read_saif
-input file_name
-instance_name name
[-target_instance instance]
[-ignore ignore_name]
[-ignore_absolute ig_absolute_name]
[-exclude exclude_file_name]
[-exclude_absolute ex_absolute_file_name]
[-names_file name_changes_log_file]
[-scale scale_value]
[-unit_base unit_value]
[-khrate khrate_value]
[-map_names]
[-auto_map_names]
[-rtl_direct]
[-verbose]
```

read_sdc

Reads in a script in Synopsys Design Constraints (SDC) format.

```
status read_sdc
file_name
[-echo]
[-syntax_only]
[-version sdc_version]
```

read_sdf

Reads leaf cell and net timing information from a file in Standard Delay Format (SDF) and uses that information to annotate the current design.

```
string read_sdf
[-load_delay net | cell]
[-path path_name]
[-min_type sdf_min | sdf_typ | sdf_max]
[-max_type sdf_min | sdf_typ | sdf_max]
[-worst]
[-min_file min_sdf_file_name]
[-max_file max_sdf_file_name]
sdf_file_name
```

read_sverilog

Reads in one or more design or library files in SystemVerilog format.

```
string read_sverilog
file_names
[-netlist]
[-rtl]
```

read_test_model

Reads a test model file.

```
list read_test_model
[-format ddc | ctl]
[-design design_name]
model_files
```

read_test_protocol

Reads a test protocol file into memory.

```
int read_test_protocol
[-verbose]
[-test_mode test_mode_name]
[-overwrite]
[-section section_name]
input_file_name
```

read_verilog

Reads in one or more design or library files in Verilog format.

```
status read_verilog
[-netlist | -rtl]
verilog_files
```

read_vhdl

Reads in one or more design or library files in VHDL format.

```
string read_vhdl
[-netlist]
file_names
```

rebuild_mw_lib

Rebuilds the Milkyway library.

```
status_value rebuild_mw_lib
libName
```

redirect

Redirects the output of a command to a file.

```
string redirect
[-append] [-tee] [-file
 | -variable
 | -channel] [-compress]
target
{command_string}
```

remove_annotated_check

Removes annotated timing check information.

```
int remove_annotated_check
-all | -from from_list | -to to_list
[-rise | -fall]
[-clock rise | fall]
[-setup] [-hold]
[-recovery] [-removal]
[-nochange_low] [-nochange_high]
```

remove_annotated_delay

Removes the annotated delay between two pins.

```
int remove_annotated_delay
-all
  | -cell_all
  | -net_all
  | -non_clock_cell_all
  | -non_clock_net_all
  | -from from_list
  | -to to_list
```

remove_annotated_transition

Removes the annotated transition at a pin.

```
int remove_annotated_transition
-all
```

remove_annotations

Removes all annotated information on the design.

```
status remove_annotations
```

remove_attribute

Removes an attribute from the specified objects.

```
collection remove_attribute
[-bus]
[-quiet]
object_list
attribute_name
```

remove_boundary_cell

Removes boundary cell configuration for the specified ports or core cells.

```
int remove_boundary_cell
-class core_wrapper | shadow_wrapper | bsd
[-ports port_list]
[-function input | output | control | bidir
| observe | input_inverted | output_inverted
| bidir_inverted]
[-core core_cell_list]
```

remove_boundary_cell_io

Removes the boundary cell IO specifications from the current boundary-scan design.

```
status remove_boundary_cell_io
[-all]
[-cell cell_list]
```

remove_bounds (Design Compiler topographical mode only)

Removes bounds from the current design.

```
int remove_bounds
[-verbose]
[-all]
[-name bound_name_list]
objects
```

remove_bsd_compliance

Removes the Compliance Ports specifications from the current boundary-scan design.

```
status remove_bsd_compliance
[-all]
[-name pattern_name]
```

remove_bsd_instruction

Removes boundary-scan instructions from the instruction list to be used by insert_dft for the current design.

```
int remove_bsd_instruction
instruction_list
```

remove_bsd_linkage_port

Removes the Linkage Ports specifications from the current boundary-scan design.

```
status remove_bsd_linkage_port
[-all]
[-port_list port_list]
```

remove_bsd_port

Removes from specified ports the attributes that identify those ports as IEEE Std 1149.1 test access ports (TAPs) in the current design.

```
int remove_bsd_port
```

remove_bsd_power_up_reset

Removes the power up reset port specifications from the current boundary-scan design.

```
status remove_bsd_power_up_reset
```

remove_buffer

Removes the buffer cells at a specified driver pin or net on a mapped design.

```
status remove_buffer
-from start_point
-net net_list
[-to end_point_list]
[-level integer]
cell_list
```

remove_bus

Removes a port bus or net bus.

```
status remove_bus
object_list
```

remove_cache

Selectively removes elements from the synthetic library cache directories.

```
int remove_cache
[-design_lib list] [-module list]
[-implementation list]
[-parameters parameter_list]
[-tech_lib list]
[-wire_load list]
[-operating_conditions list]
[-directory dir_list]
[-smaller size | -larger size]
[-netlist_only | -model_only]
[-accessed_since days
 | -accessed_beyond days]
```

remove_case_analysis

Removes the case analysis value from the specified input ports or pins.

```
string remove_case_analysis
port_or_pin_list | -all
```

remove_cell

Removes cells from the current design.

```
status remove_cell
cell_list | -all
```

remove_cell_degradation

Removes the cell_degradation attribute on specified ports or designs.

```
int remove_cell_degradation
object_list
```

remove_clock

Removes clocks from the current design.

```
int remove_clock
clock_list | -all
```

remove_clock_gating

Directs the compile -incremental, and compile_ultra -incremental commands to remove clock gating from objects clock-gated by Power Compiler.

```
integer remove_clock_gating
[-gated_registers gated_register_list]
[-min_bitwidth minsize_value]
[-gating_cells clock_gating_cells_list]
[-all]
[-no_hier]
[-verbose]
[-undo]
```

remove_clock_gating_check

Removes setup and hold checks from the specified clock gating cells.

```
int remove_clock_gating_check
[-setup]
[-hold]
[-rise]
[-fall]
object_list
```

remove_clock_groups

Removes specific exclusive or asynchronous clock groups from the current design.

```
status remove_clock_groups
-logically_exclusive
 | -asynchronous
 | -physically_exclusive
name_list | -all
```

remove_clock_latency

Removes clock latency information from the specified objects.

```
string remove_clock_latency
[-fall]
[-min]
[-max]
[-source]
[-early]
[-late]
object_list
[-rise]
```

remove_clock_sense

Removes clock sense information from the specified pins.

```
integer remove_clock_sense
[-all]
[-clocks clock_list]
pins
```

remove_clock_transition

Removes clock transition attributes on the specified clock objects.

```
int remove_clock_transition
clock_list
```

remove_clock_uncertainty

Removes clock uncertainty information previously set by the set_clock_uncertainty command.

```
string remove_clock_uncertainty
[object_list
 | -from from_clock
 | -rise_from rise_from_clock
 | -fall_from fall_from_clock
-to to_clock
 | -rise_to rise_to_clock
 | -fall_to fall_to_clock]
[-rise]
[-fall]
[-setup]
[-hold]
```

remove_congestion_options (Design Compiler topographical mode only)

Removes congestion options from the current design.

```
int remove_congestion_options
[-all] id_list
```

remove_constraint

Removes all constraint attributes, clocks, and path delay information from the current design. Note that this command will be obsolete in the next release. It will be replaced by the remove_sdc command. Please adjust your scripts accordingly.

```
int remove_constraint
-all
[-all_sdc]
```

remove_cts_scenario (Design Compiler topographical mode only)

Removes the current CTS scenario setting. This command doesn't remove the scenario itself, it only removes the CTS scenario setting.

```
Boolean remove_cts_scenario
```

remove_data_check

Removes specified data-to-data checks previously set by set_data_check.

```
string remove_data_check
-from from_object
  | -rise_from from_object
  | -fall_from from_object
-to to_object
  | -rise_to to_object
  | -fall_to to_object
[-setup | -hold]
[-clock clock]
```

remove_design

Removes a list of designs or libraries from memory.

```
status remove_design
[design_list | -designs | -all]
[-hierarchy] [-quiet]
```

remove_dft_connect

Removes existing DFT connectivity specifications.

```
status remove_dft_connect
label_name | -all
[-all]
```

remove_dft_design

Removes the specified design so that it cannot be used for DFT insertion.

```
integer remove_dft_design
-design_name design_name
  | -type design_type
  | -all
```

remove_dft_equivalent_signals

Removes all the equivalent dft signals specified with set_dft_equivalent_signals for the specified primary signal commands. Supported only in XG mode.

```
int remove_dft_equivalent_signals
primary_signal
```

remove_dft_location

Removes the DFT Hierarchy location specification for the current design.

```
integer remove_dft_location
[-type ALL | MOXIE | TCM | LOCKUP_LATCH |
PIPELINE_SI_LOGIC | PIPELINE_SE_LOGIC]
```

remove_dft_partition

Permanently removes a list of design DFT partitions associated with a design.

```
status remove_dft_partition
[list_of_partition_labels]
```

remove_dft_signal

Removes from specified ports the attributes that identify those ports as DFT signals in the current design.

```
int remove_dft_signal
-view existing_dft | spec
-port port_list
-test_mode mode_list
[-hookup_pin pin_name]
```

remove_disable_clock_gating_check

For specified cells and pins, restores clock gating checks previously disabled by the set_disable_clock_gating_check command.

```
string remove_disable_clock_gating_check
object_list
```

remove_disable_timing

Enable previously user-disabled timing arcs in the current design. It is an equivalent to set_disable_timing -restore.

```
int remove_disable_timing
object_list
[-from from_pin_name -to to_pin_name]
[-all_loop_breaking]
```

remove_dp_int_round

Remove the rounding attribute from datapath output nets.

```
int remove_dp_int_round
nets
```

remove_driving_cell

Removes driving cell attributes from the specified input or inout ports of the current design.

```
int remove_driving_cell
[port_list]
```

remove_from_collection

Removes objects from a collection, resulting in a new collection. The base collection remains unchanged.

```
collection remove_from_collection
base_collection
object_spec
```

remove_from_rp_group (Design Compiler topographical mode only)

Removes an item (cell, relative placement group, or keepout) from the specified relative placement groups.

```
status remove_from_rp_group
rp_groups
-leaf cell_name
-hierarchy group_name
[-instance instance_name]
-keepout keepout_name
```

remove_generated_clock

Removes a generated_clock object.

```
string remove_generated_clock
-all |
clock_list
```

remove_host_options

Remove any *-max_cores* specification set by the set_host_options command.

```
status remove_host_options
```

remove_ideal_latency

Removes ideal latency information from the specified objects.

```
string remove_ideal_latency
[-rise | -fall] [-min | -max] object_list |
-all
```

remove_ideal_net

Restores the ideal nets set by the set_ideal_net or set_ideal_network -no_propagate command to their initial nonideal state from the specified nets in the current design.

```
status remove_ideal_net
net_list
```

remove_ideal_network

Removes a set of ports or pins in an ideal network in the current design. Cells and nets in the transitive fanout of the specified objects are no longer treated as ideal.

```
int remove_ideal_network
object_list | -all
```

remove_ideal_transition

Removes ideal transition information from the specified objects.

```
string remove_ideal_transition
[-rise | -fall] [-min | -max] object_list |
-all
```

remove_ignored_layers (Design Compiler topographical mode only)

Removes ignored routing layers in congestion analysis and RC estimation. This command is supported only in topographical mode.

```
int remove_ignored_layers
list_of_layers
[-all]
[-min_routing_layer]
[-max_routing_layer]
```

remove_input_delay

Removes input delay on pins or input ports.

```
int remove_input_delay
[-clock clock] [-clock_fall]
[-level_sensitive]
[-rise]
[-fall]
[-max]
[-min]
port_pin_list
```

remove_isolate_ports

Removes the specified ports from the list of ports that are isolated in the current design.

```
int remove_isolate_ports
port_list
```

remove_isolation_cell

Removes specified isolation cell or cells from design.

```
int remove_isolation_cell
[-force]
-object_list cells
```

remove_level_shifters

Removes all of the level shifters from the design.

```
integer remove_level_shifters
[-force]
```

remove_license

Removes a licensed feature.

```
status remove_license
feature_list
```

remove_multibit

Removes the multibit components from the current design. Removes or detaches cells from the multibit components in a design.

```
status remove_multibit
object_list
```

remove_net

Removes nets from the *current design*.

```
int remove_net
net_list | -all
[-only_physical]
```

remove_nominal_power_state

Removes a list of power states.

```
status remove_nominal_power_state
[power_state_name_list]
```

remove_operand_isolation

Removes the isolation logic inserted by Power Compiler during the operand isolation step.

```
status remove_operand_isolation
[-from from_list]
[-to to_list]
```

remove_output_delay

Removes output delay on pins or output ports.

```
int remove_output_delay
[-clock clock [-clock_fall]
[-level_sensitive]]
[-rise]
[-fall]
[-max]
[-min]
port_pin_list
```

remove_pass_directories

Removes the data directories associated with the specified passes.

```
int remove_pass_directories
pass_list
```

remove_pin_map

Removes a design port-to-package pin mapping for a boundary-scan design.

```
int remove_pin_map
package_name
```

remove_pin_name_synonym

Removes pin name synonym definitions.

```
status remove_pin_name_synonym
[-all]
[synonym_list]
```

remove_port

Removes ports from the current design or its subdesign.

```
int remove_port
port_list
```

remove_power_domain

Removes the specified power domains.

```
status remove_power_domain
domain_name | -all
```

remove_power_net_info

Deletes power net info.

```
int remove_power_net_info
name
-all
```

remove_preferred_routing_direction (Design Compiler topographical mode only)

Removes the preferred routing direction for the given routing layer(s).

```
string remove_preferred_routing_direction
-layers list_of_layers
```

remove_propagated_clock

Removes a propagated clock specification.

```
string remove_propagated_clock
object_list
```

remove_rp_group_options (Design Compiler topographical mode only)

Removes relative placement (RP) group attributes from the specified relative placement groups.

```
collection remove_rp_group_options
rp_groups
[-ignore]
[-x_offset]
[-y_offset]
[-compress]
```

remove_rp_groups (Design Compiler topographical mode only)

Removes a list of relative placement (RP) groups.

```
status remove_rp_groups
rp_groups | -all
[-hierarchy]
[-quiet]
```

remove_rtl_load

Removes previously-set capacitance and resistance RTL load values from pins, ports, and nets.

```
int remove_rtl_load
-all | pin_net_list
```

remove_scaling_lib_group

Removes any previously specified scaling_lib_group from the current design, or from a subdesign.

```
status remove_scaling_lib_group
[-object_list objects]
```

remove_scan_group

Removes an existing scan group specification previously specified using the set_scan_group command.

```
status remove_scan_group
scan_group_name
```

remove_scan_link

Removes a scan link specification for the current design.

```
int remove_scan_link
scan_link_name
{Wire | Lockup}
[-test_mode test_mode]
```

remove_scan_path

Remove the scan path specification for the current design in set_scan_path.

```
int remove_scan_path
[-chain scan_chain_name]
[-view existing_dft | spec]
[-test_mode test_mode]
```

remove_scan_register_type

Removes existing scan register types, previously set by set_scan_register_type, from specified cells or from the current design.

```
integer remove_scan_register_type
[cell_or_design_list]
```

remove_scan_replacement

Removes the table entries specified thru set_scan_replacement, a table of one-to-one mappings of flops to their equivalent scan flops from the target library that are to be used by insert_scan or compile -scan when scan replacing cell instances.

```
int remove_scan_replacement
[non_scan_seq_cell_list]
```

remove_scan_suppress_toggling

Removes the existing user specifications that were provided through the set_scan_suppress_toggling command in terms of a list of scan flip-flops to be gated by the insert_dft command.

```
status remove_scan_suppress_toggling
```

remove_scenario (Design Compiler topographical mode only)

Removes a scenario from memory.

```
status remove_scenario
[scenario_name | -all]
```

remove_sdc

Removes all Synopsys Design Constraints (SDC).

```
int remove_sdc
[-keep_parasitics]
```

remove_target_library_subset

Removes target library subset constraints (including both the target library subset specified by library_list option and -milkyway_reflibs option) from root design or from specified instances.

```
int remove_target_library_subset
[-object_list cells]
[-top]
```

remove_test_mode

Removes the mode declared by the define_test_mode command.

```
int remove_test_mode
test_mode_label
```

remove_test_model

Permanently removes the test model associated with a design.

```
int remove_test_model
[-design design_name]
```

remove_test_point_element

Removes the test point element specification for a particular test point type and a list of pin objects for the current design.

```
int remove_test_point_element
list_of_design_pin_objects
[-type control_0 | control_z0 | control_1 |
control_z1 | control_01 | control_z01 |
force_0 | force_z0 | force_1 | force_z1 |
force_01 | force_z01 | observe]
```

remove_test_protocol

Removes a test protocol from memory for the current design.

```
int remove_test_protocol
[-design design_name]
[-test_mode test_mode_name]
```

remove_unconnected_ports

Removes unconnected ports or pins from cells, references, and subdesigns.

```
int remove_unconnected_ports
cell_list
[-blast_buses]
```

remove_upf

Removes the UPF constraints from the design. This command is only supported in dc_shell.

```
status remove_upf
```

remove_user_attribute

Removes a user-specified attribute from a design or library object.

```
list remove_user_attribute
object_list
attribute_name
[-bus]
[-quiet]
```

remove_voltage_area (Design Compiler topographical mode only)

Removes voltage areas from the current design.

```
int remove_voltage_area
-all | patterns
[-name list]
```

remove_wire_load_min_block_size

Removes the wire_load_min_block_size attribute from the current design.

```
status remove_wire_load_min_block_size
```

remove_wire_load_model

Removes wire load model attributes from designs, ports, hierarchical cells, or the specified cluster of the current design.

```
int remove_wire_load_model
[-min] [-max]
[object_list]
```

```
int remove_wire_load_model
[-min] [-max]
[object_list]
```

remove_wire_load_selection_group

Removes wire load model selection group from designs and cells, or from a specified cluster of the current design.

```
int remove_wire_load_selection_group
[-min] [-max]
[object_list]
int remove_wire_load_selection_group
[-min] [-max]
[object_list]
```

rename

Rename or delete a command.

```
string rename
oldName newName
string oldName
```

rename_design

Renames design in memory, or moves a list of designs to a file.

```
status rename_design
design_list
[target_name]
[-prefix prefix_name]
[-postfix postfix_name]
[-update_links]
```

rename_mw_lib

Renames a Milkyway library.

```
status_value rename_mw_lib
-from lib_name
-to lib_name
```

replace_clock_gates

Replaces manually inserted clock gates with Power Compiler clock gates.

```
int replace_clock_gates
[-global]
[-no_hier]
```

replace_synthetic

Implements all synthetic library parts of a design using generic logic.

```
int replace_synthetic
[-ungroup]
```

report_ahfs_options (Design Compiler topographical mode only)

Writes a report about the automatic high-fanout synthesis (AHFS) options.

```
integer report_ahfs_options
```

report_annotated_check

Displays all annotated timing checks on the current design.

```
int report_annotated_check
[-nosplit]
```

report_annotated_delay

Displays all annotated delays on cells and nets of the current design.

```
int report_annotated_delay
[-cell] [-net] [-nosplit] [-summary]
```

report_annotated_transition

Displays annotated transitions on all pins of the current design.

```
int report_annotated_transition
-nosplit
```

report_app_var

Shows the application variables.

```
string report_app_var
[-verbose]
[-only_changed_vars]
[pattern]
```

report_area

Displays area information for the current design or instance.

```
integer report_area
[-nosplit]
[-physical]
[-hierarchy]
```

report_attribute

Displays attributes and their values associated with a cell, net, pin, port, instance, or design.

```
string report_attribute
[-net]
[-cell]
[-design]
[-reference]
[-hierarchy]
[-instance]
[-pin]
[-port]
[-nosplit]
[object_list]
```

report_auto_ungroup

Displays information about the cell hierarchies that have been ungrouped using the compile command with either the -auto_ungroup area or the -auto_ungroup delay option.

```
int report_auto_ungroup
[-nosplit]
[-full]
```

report_autofix_configuration

Displays options set by the set_autofix_configuration command.

```
integer report_autofix_configuration
[-type all | clock | set | reset |
xpropagation | internal_bus | external_bus]
```

report_autofix_element

Displays options set by the set_autofix_element command.

```
status report_autofix_element
[-type all | clock | set | reset |
xpropagation | internal_bus | external_bus]
[list_of_design_objects]
```

report_boundary_cell

Reports boundary cell configuration specified for the current design.

```
status report_boundary_cell
```

report_boundary_cell_io

Displays options set by the set_boundary_cell_io command.

```
status report_boundary_cell_io
```

report_bounds (Design Compiler topographical mode only)

Reports bounds in the design.

```
int report_bounds
-all | bound | -name name_list
[-verbose]
```

report_bsd_compliance

Displays options set by the set_bsd_compliance command.

```
integer report_bsd_compliance
```

report_bsd_instruction

Displays options set by the set_bsd_instruction command.

```
status report_bsd_instruction
[-view spec | existing_dft]
[-instruction list_of_instruction_names]
```

report_bsd_linkage_port

Displays options set by the set_bsd_linkage_port command.

```
integer report_bsd_linkage_port
```

report_bsd_power_up_reset

Displays options set by the set_bsd_power_up_reset command.

```
integer report_bsd_power_up_reset
```

report_buffer_tree

Reports the buffer tree and its level information at the given driver pin.

```
status report_buffer_tree
[-from start_point_list | -net net_list]
[-break_points]
[-depth max_depth]
[-connections]
[-hierarchy]
[-physical]
[-nosplit]
```

report_buffer_tree_qor

Displays quality related properties of the buffer trees at the given driver pins.

```
int report_buffer_tree_qor
[-from list_of_driving_pins_or_nets]
```

report_bus

Lists the bused ports and nets in the current instance or in the current design.

```
integer report_bus
[-nosplit]
```

report_cache

Reports on the contents of synthetic library caches.

```
int report_cache
[-design_lib list]
[-module list]
[-implementation list]
[-parameters parameter_list]
[-tech_lib list]
[-wire_load list]
[-operating_conditions list]
[-directory dir_list]
[-smaller size | -larger size]
[-accessed_since days
 | -accessed_beyond days]
[-netlist_only | -model_only]
[-sort_largest | -sort_oldest |
-sort_cache_key]
[-statistics]
```

report_case_analysis

Reports case analysis on ports or pins.

```
string report_case_analysis
[-all] [-nosplit]
```

report_cell

Displays information about cells in the current instance or in the current design.

```
status report_cell
[-nosplit]
[-connections]
[-verbose]
[-physical]
[-only_physical]
[-significant_digits digits]
[cell_list]
```

report_check_library_options

Reports the values or the status of the options set by the set_check_library_options command. The report_check_library_options command provides information about options used for checking between logical libraries, options used for checking between physical libraries, and options used for checking between logical libraries and physical libraries.

```
status report_check_library_options
[-physical]
[-logic_vs_physical]
[-logic]
[-default]
```

report_clock

Displays clock-related information on the current design.

```
status report_clock
[-attributes] [-skew] [-nosplit] [-groups]
[-scenario
```

report_clock_gating

Reports information about clock gating performed by Power Compiler.

```
status report_clock_gating
[-no_hier]
[-verbose]
[-gated]
[-ungated]
[-gating_elements]
[-only cell_list]
[-nosplit]
[-physical]
[-multi_stage]
[-style]
[-structure]
[-scenario scenario_list]
```

report_clock_gating_check

Prints a report of the clock gating checks.

```
string report_clock_gating_check
[-nosplit]
[-significant_digits digits]
[instance_list]
```

report_clock_timing

Reports the timing attributes of clock networks.

```
string report_clock_timing
-type report_type
[-clock clock_list]
[-from_clock from_clock_list]
[-to_clock to_clock_list]
[-to to_list]
[-from from_list]
[-setup] | [-hold]
[-launch] | [-capture]
[-rise] | [-fall]
[-min] | [-max]
[-nworst worst_entries]
[-greater_than lower_limit]
[-lesser_than upper_limit]
[-slack_lesser_than slack_upper_limit]
[-include_uncertainty_in_skew]
[-verbose]
[-show_clocks]
[-nosplit]
[-significant_digits digits]
[-nets]
[-capacitance]
[-attributes]
[-physical]
[-max]
[-fall]
[-capture]
[-hold]
```

report_clock_tree

Reports structural and timing characteristics of a compiled clock tree.

```
status report_clock_tree
[-clock_trees clock_tree_list]
[-summary]
[-structure]
[-drc_violators]
[-settings]
[-exceptions [-show_all_sinks]]
[-from from_list | -to to_list]
[-operating_condition condition]
[-level_info]
[-high_fanout_net net_or_pin_list
 | -premesh
 | -postmesh]
[-nosplit]
[-all_drc_violators]
[-partial_structure_within_exceptions]
[-skew_group skew_groups_string]
```

report_compile_options

Displays information about the compile options for the design of the current instance, if set; or for the current design otherwise.

```
int report_compile_options
[-nosplit]
```

report_congestion (Design Compiler topographical mode only)

Reports the congestion statistics.

```
status report_congestion
```

report_congestion_options (Design Compiler topographical mode only)

Reports congestion options in the design.

```
integer report_congestion_options
[-all]
id_list
```

report_constraint

Displays constraint-related information about a design.

```
status report_constraint
[-all_violators]
[-verbose]
[-significant_digits digits]
[-max_area]
[-max_delay]
[-critical_range]
[-min_delay]
[-max_capacitance]
[-min_capacitance]
[-max_transition]
[-max_fanout]
[-cell_degradation]
[-min_porosity]
[-max_dynamic_power]
[-max_leakage_power]
[-max_net_length]
[-connection_class]
[-multiport_net]
[-nosplit]
[-max_toggle_rate]
[-max_total_power]
[-scenario scenario_list]
```

report_crpr

Reports the clock reconvergence pessimism calculated between specified register clock pins or ports.

```
status report_crpr
-from from_latch_clock_pin
-to to_latch_clock_pin
[-from_clock from_clock]
[-to_clock to_clock]
[-setup | -hold]
[-significant_digits digits]
```

report_delay_calculation

Displays the actual calculation of a timing arc delay value for a cell or net.

```
status report_delay_calculation
-min
-max
-from from_pin
-to to_pin
[-nosplit]
[-crosstalk]
[-from_rise_transition from_rise_value]
[-from_fall_transition from_fall_value]
```

report_delay_estimation_options (Design Compiler topographical mode only)

Reports the parameters that influence delay estimation. This command is supported only in topographical mode.

```
status report_delay_estimation_options
```

report_design

Displays attributes of the current design.

```
int report_design
[-nosplit]
[-physical]
```

report_design_lib

Lists the design units contained in the specified libraries.

```
int report_design_lib
[-libraries] [-designs] [-architectures]
[-packages] [library_list]
```

report_dft_configuration

Displays the options specified by the set_dft_configuration command.

```
integer report_dft_configuration
```

report_dft_connect

Reports the existing DFT connectivity specifications.

```
status report_dft_connect
```

report_dft_design

Reports all user-specified DFT designs.

```
int report_dft_design
[-all]
[-type design_type_name]
[-design_name design_name]
```

report_dft_drc_rules

Reports the severity of the DRC violations for the cells in the design.

```
integer report_dft_drc_rules
[-all]
[-violation drc_error_ID]
[-cell cell_list]
```

report_dft_equivalent_signals

Reports all of the equivalent DFT signals specified with set_dft_equivalent_signals command.

```
integer report_dft_equivalent_signals
```

report_dft_insertion_configuration

Displays options set by set_dft_insertion_configuration command.

```
int report_dft_insertion_configuration
```

report_dft_location

Reports the DFT Hierarchy location specification for the current design.

```
integer report_dft_location
[-all]
```

report_dft_partition

Displays design partition information attached to a design.

```
status report_dft_partition
```

report_dft_signal

Displays options specified by the set_dft_signal command.

```
integer report_dft_signal
[-view spec | existing_dft]
[-test_mode list_of_mode_names | all]
[-port list_of_port_names]
[-type type]
```

report_direct_power_rail_tie

Reports all the library pins on which the direct_power_rail_tie attribute is set to true.

```
int report_direct_power_rail_tie
```

report_disable_timing

Reports disabled timing arcs in the current design.

```
string report_disable_timing
[-nosplit]
```

report_dp_smartgen_options

Displays datapath strategies available to the current design.

```
integer report_dp_smartgen_options
```

report_extraction_options (Design Compiler topographical mode only)

Reports the options that influence postroute extraction.

```
status report_extraction_options
[-scenario scenario_list]
```

report_fault

Report the fault coverage for the design.

```
int report_fault
[-design design_name]
[-exec exec_name]
```

report_fsm

Displays state-machine attributes and information for the design of the current instance or for the current design.

```
integer report_fsm
[-nosplit]
```

report_hierarchy

Displays the reference hierarchy of the current instance or the current design.

```
integer report_hierarchy
[-nosplit]
[-full]
[-noleaf]
```

report_host_options

Prints a report of multi-CPU processing options as defined by the *set_host_options* command.

```
int report_host_options
```

report_ideal_network

Displays information about ports, pins, nets, and cells on ideal networks in the current design.

```
int report_ideal_network
[-net]
[-cell]
[-load_pin]
[-timing]
[object_list]
```

report_ignored_layers (Design Compiler topographical mode only)

Reports the routing layers that are ignored during congestion analysis and RC estimation. This command is supported only in topographical mode.

```
status report_ignored_layers
```

report_ilm

Reports information about the specified ILM instance.

```
status report_ilm
[ilm_list]
```

report_interclock_relation

Displays common multiple clock period between clocks of different periods.

```
report_interclock_relation
[-from clock_name]
[-to clock_name]
[-nosplit]
[-significant_digits digits]
```

report_internal_loads

Displays internal loads on the nets in the current design.

```
int report_internal_loads
[-nosplit]
```

report_isolate_ports

Displays the status of port isolation on ports on which isolation was requested.

```
int report_isolate_ports
[-nosplit]
```

report_isolation_cell

Displays information about isolation cells in the current scope. This command is supported only in UPF mode.

```
status report_isolation_cell
[isolation_cells]
[-domain power_domains]
[-isolation_strategy
isolation_strategy_names]
[-ports pins_ports]
[-verbose]
[-nosplit]
```

report_level_shifter

Displays information about level shifter cells in the current scope. This command is supported only in UPF mode.

```
status report_level_shifter
[level_shifter_cells]
[-domain power_domains]
[-verbose]
[-nosplit]
```

report_lib

Displays information about technology, symbol libraries, or physical libraries.

```
int report_lib
[-all]
[-ccs_recv]
[-em]
[-fpga]
[-full_table]
[-k_factors]
[-power]
[-power_label]
[-routing_rule]
[-rwm]
[-table]
[-timing]
[-timing_arcs]
[-timing_label]
[-user_defined_data]
[-vhdl_name]
[-yield]
[-switch]
[-pg_pin]
[-char]
[-operating_condition]
[-op_cond_name op_cond_name]
[cell_list]
library_name
```

report_logicbist_configuration

Displays options specified by the set_logicbist_configuration command.

```
integer report_logicbist_configuration
[-test_mode test_mode_names | all]
```

report_mode

Prints a report of the instance modes.

```
string report_mode
[-nosplit]
[instance_list]
```

report_multibit

Displays information about multibit components in the current design or the subdesign.

```
status report_multibit
[-nosplit]
[object_list]
```

report_mw_lib

Displays information about a Milkyway library.

```
status_value report_mw_lib
[-unit_range]
[-mw_reference_library]
mw_lib
```

report_name_rules

Reports the values of name rules.

```
int report_name_rules
[name_rules]
```

report_names

Reports potential name changes of ports, cells, and nets in a design.

```
int report_names
[-rules name_rules]
[-hierarchy]
[-dont_touch designs_list]
[-nosplit]
[-original]
```

report_net

Reports net information for the design of the current instance or for the current design.

```
status report_net
[-nosplit]
[-noflat]
[-transition_times]
[-only_physical]
[-verbose]
[-cell_degradation]
[-min]
[-connections]
[-physical]
[net_list]
[-significant_digits digits]
[-max_toggle_rate]
[-scenario scenario_list]
```

report_net_changes

Reports net changes that occurred during some IC Compiler optimizations, such as place_opt, create_buffer_tree, and route_opt.

```
int report_net_changes
[-verbose]
```

report_net_fanout

Displays net fanout or buffer tree information for the current design.

```
status report_net_fanout
[-nosplit]
[-high_fanout]
[-threshold lower]
[-bound upper]
[-verbose]
[-connections]
[-physical]
[-min]
[-tree [-depth level]]
[net_list]
```

report_nominal_power_state_template

Reports the current power state template.

```
integer report_nominal_power_state_template
```

report_operand_isolation

Reports the status of operand isolation cells in the current design.

```
status report_operand_isolation
[-instances]
[-isolated_objects]
[-unisolated_objects]
[-all_objects]
[-verbose]
[-no_hier]
[-nosplit]
[object_list]
```

report_operating_conditions

Display a specific or all the operating conditions in a library.

```
int report_operating_conditions
-library library_name
[-name op_cond_name]
```

report_partitions

Lists the hierarchical designs and their associated attributes and relative size (estimated in RTL).

```
int report_partitions
[-nosplit]
```

report_pass_data

Reports the data files that are available for a design created by Automated Chip Synthesis.

```
status report_pass_data
[-hierarchy]
[-pass_list pass_list]
[design]
```

report_path_budget

Displays budgeting information about a design.

```
integer report_path_budget
[-to to_list]
[-from from_list]
[-through through_list]
[-nworst paths_per_endpoint]
[-max_paths max_path_count]
[-input_pins]
[-nets]
[-transition_time]
[-significant_digits digits]
[-nosplit]
[-all]
[-verbose]
```

report_path_group

Reports information about path groups in the current design.

```
status report_path_group
[-nosplit]
[-expanded]
[-scenario scenario_list]
```

report_physical_constraints (Design Compiler topographical mode only)

Reports the current physical constraints setting. This command is supported only in topographical mode.

```
int report_physical_constraints
[-site_row]
[-pre_route]
```

report_pin_map

Displays package pin map information set by the read_pin_map command.

```
status report_pin_map
```

report_pin_name_synonym

Reports pin name synonym definitions.

```
status report_pin_name_synonym
[-nosplit]
```

report_port

Displays information about ports of the current instance or the current design.

```
integer report_port
[-drive]
[-verbose]
[-physical]
[-only_physical]
[-nosplit]
[-significant_digits digits]
[port_list]
```

report_power

Calculates and reports dynamic and static power for a design or instance.

```
int report_power
[-net]
[-cell]
[-only cell_or_net_list]
[-hier]
[-hier_level level_value]
[-verbose]
[-cumulative]
[-flat]
[-exclude_boundary_nets]
[-include_input_nets]
[-analysis_effort low | medium | high]
[-nworst number]
[-sort_mode mode]
[-histogram [-exclude_leq le_val
 | -exclude_geq ge_val]]
[-nosplit]
[-scenario scenario_list]
```

report_power_calculation

Displays the calculation of the internal power for a pin, the leakage power for a cell, or the switching power for a net.

```
int report_power_calculation
pin_cell_or_net_list
[-state_condition boolean_eq_of_pins |
default | all]
[-path_source pin_name | default | all]
[-rise]
[-fall]
[-verbose]
[-nosplit]
```

report_power_domain

Reports information about the specified power domain.

```
status report_power_domain
[power_domains]
```

report_power_gating

Reports the power gating style of retention registers in the design.

```
int report_power_gating
[cell_or_design_list]
[-missing]
[-unconnected]
```

report_power_net_info

Reports the power net information for the current design.

```
integer report_power_net_info
```

report_power_pin_info

Reports the power pin information for technology library cells or leaf cells. In UPF mode, the command reports power pin information only for instantiated cells and not the library cells.

```
status report_power_pin_info
object_list
```

report_power_state_table

Reports the legal power states in the current design.

```
integer report_power_state_table
[-power_nets power_net_list]
[-compress]
[-trace_name]
[-significant_digits digit]
[-width]
[-column_space column_space]
```

report_power_switch

Reports all of the specified power switches. This command is supported only in UPF mode.

```
status report_power_switch
[power_switch_name]
[-verbose]
```

report_preferred_routing_direction (Design Compiler topographical mode only)

Reports the preferred routing direction for all routing layers.

```
string report_preferred_routing_direction
```

report_pst

Report power states in current design (UPF mode only).

```
int report_pst
[-verbose]
[-supplies supply_list]
[-scope instance_name]
[-power_nets supply_net_list]
[-compress]
[-trace_name]
[-significant_digits digit]
[-width line_width]
[-column_space column_space]
[-derived]
```

report_qor

Displays QoR information and statistics for the current design.

```
status report_qor
[-significant_digits digits]
[-scenario scenario_list]
```

report_reference

Displays information about references in the current instance or in the current design.

```
integer report_reference
[-nosplit]
[-hierarchy]
```

report_resources

Lists the resources and datapath blocks used in the design of the current instance or in the current design.

```
integer report_resources
[-nosplit]
[-hierarchy]
```

report_retention_cell

Displays information about retention cells in the current scope. This command is supported only in UPF mode.

```
status report_retention_cell
[retention_cells]
[-domain power_domains]
[-retention_strategy
retention_strategy_names]
[-verbose]
```

report_rp_group_options (Design Compiler topographical mode only)

Reports relative placement group attributes on the specified relative placement groups.

```
status report_rp_group_options
rp_groups
```

report_saif

Reports statistics on the switching activity annotation on the current design or instance.

```
int report_saif
[-hier]
[-flat]
[-type rtl | gate]
[-rtl_saif]
[-missing]
[-only cell_or_net_list]
[-annotated_flag]
```

report_scan_chain

Reports the scan chains defined on the current design.

```
status report_scan_chain
```

report_scan_compression_configuration

Displays options specified by the set_scan_compression_configuration command or the reset_scan_compression_configuration command.

```
status report_scan_compression_configuration
[-test_mode test_mode_names]
```

report_scan_configuration

Displays options specified by the set_scan_configuration command.

```
integer report_scan_configuration
[-test_mode mode_names | all]
```

report_scan_group

Reports all scan group that were specified using the set_scan_group command.

```
int report_scan_group
scan_group_names
```

report_scan_link

Reports the scan links specified with the set_scan_link command.

```
int report_scan_link
-test_mode mode_name
[link_type]
```

report_scan_path

Displays scan paths and scan cells specified by the set_scan_path command and displays scan paths inserted by the insert_dft command.

```
integer report_scan_path
[-view spec | existing_dft]
[-chain chain_name | all]
[-cell chain_name | all]
[-test_mode list_of_mode_names | all]
```

report_scan_register_type

Displays test-related information about the current design.

```
integer report_scan_register_type
```

report_scan_replacement

Displays the scan replacement table specified by the set_scan_replacement command.

```
integer report_scan_replacement
```

report_scan_state

Displays the scan state of the current design.

```
status report_scan_state
```

report_scan_suppress_toggling

Reports on the user specifications that were provided through the set_scan_suppress_toggling command in terms of the list of scan flip-flops to be gated by the insert_dft command.

```
status report_scan_suppress_toggling
```

report_scenarios (Design Compiler topographical mode only)

Reports scenarios setup information for multi-scenario design.

```
int report_scenarios
```

report_supply_net

Reports all the supply nets in the current scope. This command is supported only in UPF mode.

```
status report_supply_net
[supply_net_name]
[-include_exception]
```

report_supply_port

Reports information about the supply ports in the current scope. This command is supported only in UPF mode.

```
status report_supply_port
[supply_port_name]
```

report_synlib

Displays information about synthetic libraries.

```
int report_synlib
library [module_list]
```

report_target_library_subset

Reports target library subsets on the design.

```
int report_target_library_subset
[-object_list cells]
[-top]
```

report_test_assume

Displays the value set on the pins by the set_test_assume command.

```
status report_test_assume
```

report_test_model

Displays the test model information attached to a design.

```
int report_test_model
[-design design_name]
```

report_test_point_element

Displays options specified by set_test_point_element command.

```
integer report_test_point_element
list_of_design_pin_objects
[-type control_0 | control_z0 | control_1 |
control_z1 | control_01 | control_z01 |
force_0 | force_z0 | force_1 | force_z1 |
force_01 | force_z01 | observe]
```

report_testability_configuration

Displays options specified by the set_testability_configuration command.

```
integer report_testability_configuration
[-type control | observe |
control_and_observe]
```

report_threshold_voltage_group

Reports the percentage of cells for each threshold voltage group in the design.

```
int report_threshold_voltage_group
[cell_or_design_list]
[-verbose]
```

report_timing

Displays timing information about a design.

```
status report_timing
[-to to_list]
  | -rise_to rise_to_list
  | -fall_to fall_to_list]
[-from from_list
  | -rise_from rise_from_list
  | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-path short | full | full_clock |
full_clock_expanded | only | end]
[-delay min | min_rise | min_fall | max |
max_rise | max_fall]
[-nworst paths_per_endpoint]
[-max_paths max_path_count]
[-input_pins]
[-nets]
[-transition_time]
[-crosstalk_delta]
[-capacitance]
[-attributes]
[-physical]
[-slack_greater_than greater_slack_limit]
[-slack_lesser_than lesser_slack_limit]
[-lesser_path max_path_delay]
[-greater_path min_path_delay]
[-loops]
[-true [-true_threshold path_delay]]
[-justify]
[-enable_preset_clear_arcs]
[-significant_digits digits]
[-nosplit]
[-sort_by group | slack]
[-group group_name]
[-trace_latch_borrow]
[-derate]
[-scenario scenario_list]
[-temperature]
[-voltage]
```

report_timing_derate

Reports timing derate factors for the design or specified objects.

```
string report_timing_derate
[-include_inherited]
object_list
[-nosplit]
[-scenario scenario_list]
```

report_timing_requirements

Reports timing path requirements (user attributes) and related information.

```
int report_timing_requirements
[-attributes]
[-ignored]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
[-expanded]
[-nosplit]
```

report_tlu_plus_files (Design Compiler topographical mode only)

Reports the files used for TLUPlus extraction. This command is supported only in topographical mode.

```
integer report_tlu_plus_files
[-scenario scenario_list]
```

report_transitive_fanin

Reports logic in the transitive fanin of specified sinks.

```
int report_transitive_fanin
-to sink_list
[-nosplit]
```

report_transitive_fanout

Reports logic in the transitive fanout of specified sources.

```
int report_transitive_fanout
-clock_tree | -from source_list
[-nosplit]
```

report_units

Reports the units used for resistance, capacitance, timing, leakage power, current, and voltage in the flow. The units must be consistant with the main library units.

```
string report_units
```

report_use_test_model

Reports test model usage for each subdesign under the current design.

```
status report_use_test_model
```

report_voltage_area (Design Compiler topographical mode only)

Reports the voltage areas in the design. This command is supported only in topographical mode.

```
int report_voltage_area
-all | patterns
[-name list]
[-verbose]
[-nosplit]
```

report_wire_load

Displays the characteristics of the wire load models set on a design or in a library.

```
int report_wire_load
[-design design_name]
[-name model_name]
[-libraries]
[-nosplit]
```

report_wrapper_configuration

Reports the wrapper configuration of the current design.

```
status report_wrapper_configuration
```

reset_autofix_configuration

Resets the autofix configuration on a type basis for the current design.

```
integer reset_autofix_configuration
[-type clock | set | reset | xpropagation |
internal_bus | external_bus | bidirectional]
```

reset_autofix_element

Resets the autofix configuration for a particular type and a list of design objects for the current design.

```
int reset_autofix_element
list_of_design_objects
[-type clock | set | reset | xpropagation |
internal_bus | external_bus]
```

reset_bsd_configuration

Removes the IEEE 1149.1 specifications from a boundary-scan design.

```
status reset_bsd_configuration
```

reset_clock_gate_latency

Resets all clock latency values previously specified for or applied to clock gating cells.

```
int reset_clock_gate_latency
[-clock clock_list]
```

reset_design

Removes from the current design all user-specified objects and attributes, except those defined using set_attribute.

```
status reset_design
```

reset_dft_configuration

Resets the DFT configuration for the current design specified by the set_dft_configuration command.

```
integer reset_dft_configuration
```

reset_dft_drc_rules

Resets the severity to the default value on the allowed set of violations.

```
integer reset_dft_drc_rules
[-violation drc_error_ID]
[-cell cell_list]
```

reset_dft_insertion_configuration

Resets the DFT insertion configuration for the current design.

```
int reset_dft_insertion_configuration
```

reset_logicbist_configuration

Resets the current design's logic BIST configuration previously defined by the set_logicbist_configuration command.

```
integer reset_logicbist_configuration
[-test_mode test_mode_names]
```

reset_mode

Resets the modes of the specified instances.

```
int reset_mode
[instance_list]
```

reset_nominal_power_state_template

Resets the current power state template.

```
integer reset_nominal_power_state_template
```

reset_path

Resets specified paths to single cycle timing.

```
int reset_path
[-setup | -hold]
[-rise | -fall]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
```

reset_physical_constraints (Design Compiler topographical mode only)

Resets all current physical constraints.

```
integer reset_physical_constraints
```

reset_pipeline_scan_data_configuration

Resets the pipeline scan data configuration specified by the set_pipeline_scan_data_configuration command.

```
integer
reset_pipeline_scan_data_configuration
```

reset_scan_compression_configuration

Resets the scan compression configuration for the current design.

```
int reset_scan_compression_configuration
```

reset_scan_configuration

Resets the scan configuration for the current design.

```
int reset_scan_configuration
[-test_mode mode_name]
```

reset_switching_activity

Removes the toggle rate and static probability attributes, or the maximum toggle rate attribute, from nets, pins, cells, and ports of the current design.

```
integer reset_switching_activity
-switching_activity | -max_toggle_rate |
-all
[-verbose]
[object_list]
```

reset_test_mode

Resets the test mode for the current design.

```
status reset_test_mode
```

reset_testability_configuration

Resets the testability configuration on a type basis for the current design.

```
status reset_testability_configuration
[-type control | observe |
control_and_observe]
```

reset_timing_derate

Removes all derate factors set on the current design or libraries.

```
string reset_timing_derate
```

reset_wrapper_configuration

Resets the wrapper configuration for the current design.

```
integer reset_wrapper_configuration
```

rewire_clock_gating

Changes the clock-gating cell implemented by the tool for a particular gated cell.

```
status rewire_clock_gating
[-gating_cell new_clock_gating_cell]
[-gated_objects gated_objects_list]
[-balance_fanout]
[-undo]
[-verbose]
```

rp_group_inclusions (Design Compiler topographical mode only)

Returns a collection of relative placement groups that are directly included in the specified relative placement groups.

```
collection rp_group_inclusions
[rp_groups]
```

rp_group_instantiations (Design Compiler topographical mode only)

Returns a collection of relative placement groups that are instantiated in any of the specified relative placement groups.

```
collection rp_group_instantiations
[rp_groups]
```

rp_group_references (Design Compiler topographical mode only)

Returns a collection of cells that are directly included by the specified relative placement groups.

```
collection rp_group_references
[rp_groups]
[-leaf | -instance]
```

rtl2saif

Creates a SAIF forward-annotation file starting from the top level of the design.

The rtl2saif command will be obsolete in a future release as RTL forward SAIF is now obsolete.

```
int rtl2saif
[-output file_name]
[-design design_name]
```

saif_map

Manages the SAIF name mapping mechanism for reading SAIF files.

```
string saif_map
[-start]
[-end]
[-reset]
[-report]
[-get_name]
[-set_name names]
[-add_name names]
[-remove_name names]
[-clear_name]
[-get_object_names name]
[-create_map]
[-write_map filename]
[-read_map filename]
[-type type]
[-inverted]
[-instances objects]
[-hierarchical]
[-no_hierarchical]
[-columns columns]
[-sort columns]
[-rtl_summary]
[-missing_rtl]
[-input SAIF_file]
[-source_instance SAIF_instance_name]
[-target_instance target_instance_name]
[-review]
[-preview]
[-hsep character]
[object_list]
[-verbose]
[-non_verbose]
[-nosplit]
```

save_upf

Writes out the UPF commands in the specified file. This command is supported only in UPF mode.

```
collection save_upf
upf_file_name
```

set_active_scenarios (Design Compiler topographical mode only)

Specifies which scenarios are to be active.

```
int set_active_scenarios
scenario_list | -all
```

set_ahfs_options (Design Compiler topographical mode only)

Specifies the options to be used when running automatic high-fanout synthesis (AHFS).

```
status set_ahfs_options
[-optimize_buffer_trees true | false]
[-enable_port_punching true | false]
[-no_port_punching cells]
[-default_reference references]
[-port_map_file file_name]
[-preserve_boundary_phase true | false]
[-constant_nets true | false]
[-hf_threshold integer]
[-mf_threshold integer]
[-remove_effort none | medium | high]
[-default]
```

set_always_on_strategy

Sets the always-on strategy for shutdown power domains.

```
status set_always_on_strategy
-object_list list_of_domains
-cell_type single_power | dual_power
```

set_annotated_check

Sets the setup, hold, recovery, or removal timing check value between two pins.

```
int set_annotated_check
check_value
-from from_pins
-to to_pins
-setup
 | -hold
 | -recovery
 | -removal
 | -nochange_high
 | -nochange_low
[-rise | -fall]
[-clock clock_check]
[-worst]
```

set_annotated_delay

Sets the net or cell delay value between two pins.

```
int set_annotated_delay
-net | -cell
[-load_delay load_delay_type]
[-rise | -fall] [-min] [-max] delay_value
-from from_pins -to to_pins [-worst]
```

set_annotated_transition

Sets the transition time at a given pin.

```
int set_annotated_transition
[-rise | -fall] [-min] [-max] transition
port_pin_list
```

set_app_var

Sets the value of an application variable.

```
string set_app_var
-default
var
value
```

set_aspect_ratio (Design Compiler topographical mode only)

Specifies the placement aspect ratio for the core area. This command is supported only in topographical mode.

```
int set_aspect_ratio
y_x_ratio
```

set_attribute

Sets an attribute to a specified value on the specified list of objects.

```
collection set_attribute
object_list
attribute_name
attribute_value
[-type boolean | integer | float | string]
[-bus]
[-quiet]
```

set_auto_disable_drc_nets

Sets the auto_disable_drc_net attribute on the current design, causing the specified networks to be have DRC disabled. This command was previously called set_auto_ideal_nets.

```
int set_auto_disable_drc_nets
[-default]
[-none]
[-all]
[-clock true | false]
[-constant true | false]
[-scan true | false]
```

set_autofix_configuration

Controls automatic fixing of violations when running the preview_dft or insert_dft command.

```
int set_autofix_configuration
[-type clock | set | reset | xpropagation |
internal_bus | external_bus | bidirectional]
[-test_data test_data_name]
[-method mux | gate | enable_one |
disable_all | input | output | no_disabling]
[-fix_data enable | disable]
[-fix_latch enable | disable]
[-include_elements list_of_design_objects]
[-exclude_elements list_of_design_objects]
[-control_signal control_name]
```

set_autofix_element

If one of the Autofix features is enabled (fix_clock, fix_set, fix_reset, fix_xpropagation or fix_bus), the command makes specifications on a type and design object basis during execution of preview_dft or insert_dft.

```
int set_autofix_element
list_of_design_objects
[-type clock | set | reset | xpropagation |
internal_bus | external_bus | bidirectional]
[-control_signal control_name]
[-test_data test_data_name]
[-method mux | gate | enable_one |
disable_all | input | output | no_disabling]
[-fix_data enable | disable]
[-fix_latch enable | disable]
```

set_balance_registers

Sets the balance_registers attribute on the specified designs or on the current design, so that the design is retimed during compile.

```
int set_balance_registers
[true | false]
[-design ]
[design_list]
```

set_boundary_cell

Sets the boundary cell configuration for the specified ports and core cells.

```
status set_boundary_cell
-class core_wrapper | shadow_wrapper | bsd
[-function input | output | control | bidir
| observe | input_inverted | output_inverted
| bidir_inverted | receiver_p | receiver_n |
ac_select]
[-ports port_list]
[-core_cells core_list]
[-type WC_D1 | WC_D1_S | WC_S1 | WC_S1_S |
BC1 | BC2 | BC4 | BC7 | BC8 | BC9 | AC1 | AC2
| AC7 | AC_SEL | AC_SELX | none]
[-design design_name]
[-register_io_implementation swap |
in_place]
[-use_dedicated_wrapper_clock true | false]
[-safe_state 0 | 1 | none]
[-share true | false]
[-name bcell_name]
[-shift_clk bcell_shift_clk]
```

set_boundary_cell_io

Specifies the primary inputs (PIs) and primary outputs (POs) of a boundary cell in the current design.

```
int set_boundary_cell_io
-access {pin_type pin_name}
-type boundary | wrapper
-cell shift_flop_name
```

set_boundary_optimization

Sets the boundary_optimization attribute on specified cells, references, or designs, thus allowing for optimization across hierarchical boundaries.

```
int set_boundary_optimization
obj_list
[true | false]
```

set_bsd_ac_port

Identifies the AC ports in your design.

```
int set_bsd_ac_port
-port_list list_of_ports
```

set_bsd_compliance

Specifies an IEEE 1149.1 compliant pattern for a boundary-scan design.

```
status set_bsd_compliance
-name pattern_name
-pattern signal_port_bit_value_pairs
```

set_bsd_configuration

Specifies the boundary-scan configuration for a design.

```
status set_bsd_configuration
[-asynchronous_reset true | false]
[-default_package package_name]
[-instruction_encoding binary | one_hot]
[-ir_width instruction_register_length]
[-style synchronous | asynchronous]
[-check_pad_designs none | all |
pad_design_list]
[-control_cell_max_fanout max_fanout]
[-std ieee1149.1_1993 | ieee1149.1_2001 |
ieee1149.6_2003]
[-output output_file_name]
[-rtl enable | disable]
[-fsm_width fsm_width_length]
```

set_bsd_instruction

Specifies boundary-scan instructions used by the insert_dft command for the current design or used by the check_bsd command in the verification flow.

```
status set_bsd_instruction
[-view existing_dft | spec]
instruction_list
[-code inst_code_list]
[-register register_name]
[-input_clock_condition clock_conditioning]
[-output_condition BSR | HIGHZ | NONE]
[-internal_scan pin_name]
[-capture_value capture_value_list]
[-private]
[-clock_cycles clock_cycle_list]
[-signature pattern]
[-high pin_name_list]
[-low pin_name_list]
[-time real_time]
[-excluded_bsr_condition CLAMP | NONE]
```

set_bsd_linkage_port

Identifies the linkage ports in your design.

```
int set_bsd_linkage_port
-port_list list_of_ports
```

set_bsd_port

Identifies existing ANSI/IEEE Std. 1149.1 Test Access ports of the current design, for the check_bsd command.

```
int set_bsd_port
[-tristate port_list]
[-diff_current pos_port_neg_port_pairs]
[-diff_voltage pos_port_neg_port_pairs]
```

set_bsd_power_up_reset

Specifies and characterizes the power-up reset cell for the current design.

```
integer set_bsd_power_up_reset
-cell_name cell_name
-reset_pin_name reset_pin_name
-active high | low
[-delay power_up_reset_delay]
```

set_case_analysis

Specifies that a port or pin is at a constant logic value 1 or 0, or is considered with a rising or falling transition..

```
string set_case_analysis
value
port_or_pin_list
```

set_cell_degradation

Sets the cell_degradation attribute to a specified value on specified ports or designs.

```
int set_cell_degradation
cell_degradation_value
object_list
```

set_cell_internal_power

Sets or removes the power_value attribute on the specified pins. The value represents the power consumption for a single toggle of each pin.

```
int set_cell_internal_power
[-delete_all]
pin_list
[power_value [unit]]
```

set_cell_location (Design Compiler topographical mode only)

Specifies the physical location, orientation, and dont_touch status for leaf cells. This command is supported in Design Compiler Topographical mode only.

```
int set_cell_location
object_list
-coordinates {X Y}
[-orientation string]
[-fixed]
```

set_check_library_options

Sets specific options for the check_library command for logic library versus logic library checking, logic library versus physical library checking, and for checks between physical libraries.

```
status set_check_library_options
[-cell_area]
[-cell_footprint]
[-bus_delimiter]
[-tech_consistency]
[-view_comparison]
[-same_name_cell]
[-signal_em]
[-antenna]
[-rectilinear_cell]
[-physical_only_cell]
[-phys_property {property_list}]
[-routeability]
[-tech]
[-drc]
[-scaling {scaling_types}]
[-mcmm]
[-upf]
[-compare {construct | attribute | value}]
[-tolerance {type relative_tolerance
absolute_tolerance}]
[-validate {timing}]
[-analyze nominal_vs_sigma | table_trend]
[-criteria {std_error=val_err slope=val_sl
trend=['/' | '' | '^' | 'V']}]
[-group_attribute {groups_or_attributes}]
[-report_format {csv[=csv_dir] nosplit
sort_by_cell | sort_by_group_type}]
[-physical]
[-logic_vs_physical]
[-logic]
[-reset]
[-all]
```

set_clock_gate_latency

Specifies clock network latency values to be used for clock gating cells, as a function of clock domain, clock gating stage, and fanout.

```
status set_clock_gate_latency
[-clock clock_list]
[-overwrite]
-stage cg_stage
-fanout_latency cg_fanout_list
```

set_clock_gating_check

Puts setup and hold checks on clock gating cells.

```
int set_clock_gating_check
[-setup setup_margin]
[-hold hold_margin]
[-rise]
[-fall]
[-high | -low]
[object_list]
```

set_clock_gating_registers

Forces the enabling or disabling of clock gating for specified registers in the current design, overriding all conditions necessary for automatic RTL clock gating by the compile_ultra -gate_clock command.

```
status set_clock_gating_registers
[-include_instances register_list]
[-exclude_instances register_list]
[-undo register_list]
```

set_clock_gating_style

Sets the clock-gating style for the clock-gate insertion and replacement.

```
status set_clock_gating_style
[-sequential_cell none | latch]
[-minimum_bitwidth minsize_value]
[-setup setup_value]
[-hold hold_value]
[-positive_edge_logic {cell_list |
integrated [active_low_enable]
[invert_gclk]}
[-negative_edge_logic {cell_list |
integrated [active_low_enable]
[invert_gclk]}
[-control_point none | before | after]
[-control_signal scan_enable | test_mode]
[-observation_point true | false]
[-observation_logic_depth depth_value]
[-max_fanout max_fanout_count]
[-num_stages num_stages_count]
[-no_sharing]
[-gicg_pos_cell {[cell_library/]cell_name}]
[-gicg_neg_cell {[cell_library/]cell_name}]
[-gicg_pos_auto {[cell_library/]cell_name}]
[-gicg_neg_auto {[cell_library/]cell_name}]
```

set_clock_groups

Specifies clock groups that are mutually exclusive or asynchronous with each other in a design so that the paths between these clocks are not considered during the timing analysis.

```
Boolean set_clock_groups
-physically_exclusive
 | -logically_exclusive
 | -asynchronous
[-allow_paths]
[-name name]
-group clock_list
```

set_clock_latency

Specifies clock network latency.

```
string set_clock_latency
[-rise]
[-fall]
[-min]
[-max]
[-source]
[-early]
[-late]
[-clock clock_list]
delay
object_list
```

set_clock_sense

Specifies the clock sense (with respect to the clock source) propagating forward from the specified pins.

```
string set_clock_sense
[-stop_propagation]
[-positive]
[-negative]
[-pulse {rise_triggered_high_pulse |
rise_triggered_low_pulse |
fall_triggered_high_pulse |
fall_triggered_low_pulse}]
[-clocks clocks]
pins
```

set_clock_transition

Sets clock transition attributes on clock objects.

```
int set_clock_transition
transition
[-rise | -fall]
[-min]
[-max]
clock_list
```

set_clock_uncertainty

Specifies the uncertainty (skew) of specified clock networks.

```
string set_clock_uncertainty
[object_list
 | -from from_clock
 | -rise_from rise_from_clock
 | -fall_from fall_from_clock
-to to_clock
 | -rise_to rise_to_clock
 | -fall_to fall_to_clock]
[-rise]
[-fall]
[-setup]
[-hold]
uncertainty
```

set_combinational_type

Sets attributes on cell instances to specify which combinational cells from the target library are to be used by compile.

```
int set_combinational_type
-replacement_gate replacement_gate
[cell_list]
```

set_compile_directives

Controls the application of high-level optimization operations on cells, references, designs, and library cells.

```
status set_compile_directives
object_list
[-delete_unloaded_gate true | false]
[-constant_propagation true | false]
[-local_optimization true | false]
[-critical_path_resynthesis true | false]
```

set_compile_partitions

Specifies the compile partitions for the current design.

```
status set_compile_partitions
-level level
  | -designs design_list
  | -all
  | -auto
[-force]
[-no_reset]
```

set_congestion_options (Design Compiler topographical mode only)

Sets options for congestion optimization.

```
int set_congestion_options
[-max_util value]
[-layer name]
[-availability value]
[-coordinate {X1 Y1 X2 Y2}]
```

set_connection_class

Sets the connection class value on ports.

```
status set_connection_class
connection_class_value
object_list
```

set_context_margin

Specifies the margin by which to tighten or relax constraints.

```
string set_context_margin
[-percent]
[-relax]
[-min]
[-max]
value
[object_list]
```

set_cost_priority

Sets the cost_priority attribute to a specified value on the current design.

```
int set_cost_priority
[-default]
[-delay]
cost_list
[-design_rules]
[-min_delay]
```

set_critical_range

Sets the critical_range attribute to a specified value on a list of designs.

```
int set_critical_range
range_value designs
```

set_cts_scenario (Design Compiler topographical mode only)

Sets the specified scenario as the clock tree synthesis scenario.

```
string set_cts_scenario
[scenario_name]
```

set_current_command_mode

```
string set_current_command_mode
-mode command_mode | -command command
```

set_data_check

Sets data-to-data checks using the specified values of setup and hold time.

```
string set_data_check
-from from_object
 | -rise_from from_object
 | -fall_from from_object
-to to_object
 | -rise_to to_object
 | -fall_to to_object
[-setup | -hold]
[-clock clock_object]
[check_value]
```

set_datapath_optimization_effort

Sets the datapath_optimization_effort attribute on specified designs, cells, or references, indicating the level of datapath optimization during compile_ultra.

```
status set_datapath_optimization_effort
object_list
effort_level
```

set_default_drive

Sets the default driving strength for specified objects, to be used by Top-Down Environmental Propagation (TDEP).

```
status set_default_drive
[-min]
[-max]
[-rise]
[-fall]
[-none]
[resistance]
[cell_or_pin_list]
```

set_default_driving_cell

Sets the default driving cell for specified objects, to be used by Top-Down Environmental Propagation (TDEP).

```
int set_default_driving_cell
[-lib_cell lib_cell_name]
[-library lib]
[-rise] [-fall]
[-pin pin_name]
[-from_pin from_pin_name]
[-dont_scale] [-no_design_rule]
[-multiply_by factor] [-none]
cell_or_pin_list
```

set_default_fanout_load

Sets the default fanout load to be used by Top-Down Environmental Propagation (TDEP).

```
status set_default_fanout_load
[-none]
[fanout_load_value]
[cell_or_pin_list]
```

set_default_input_delay

Sets the value of the input delay as a percentage of the clock period to be assigned during environment propagation.

```
int set_default_input_delay
[-none]
percent_delay
[cell_or_pin_list]
```

set_default_load

Sets the default load to be used by Top-Down Environmental Propagation (TDEP).

```
status set_default_load
[-min]
[-max]
[-pin_load]
[-wire_load]
[-none]
[value]
[cell_or_pin_list]
```

set_default_output_delay

Sets the output delay as a percentage of the clock period to be assigned during environment propagation.

```
int set_default_output_delay
[-none]
percent_delay
[cell_or_pin_list]
```

set_delay_calculation

Defines the delay model used to compute a timing arc delay value for a cell or net.

```
int set_delay_calculation
[-arnoldi]
[-elmore]
[-clock_arnoldi]
```

set_delay_estimation_options (Design Compiler topographical mode only)

Sets the parameters that influence preroute delay estimation. This command is supported only in topographical mode.

```
int set_delay_estimation_options
[-min_unit_horizontal_capacitance float]
[-min_unit_vertical_capacitance float]
[-min_unit_horizontal_resistance float]
[-min_unit_vertical_resistance float]
[-max_unit_horizontal_capacitance float]
[-max_unit_vertical_capacitance float]
[-max_unit_horizontal_resistance float]
[-max_unit_vertical_resistance float]
[-min_unit_horizontal_capacitance_scaling_fa
ctor float]
[-min_unit_vertical_capacitance_scaling_fact
or float]
[-min_unit_horizontal_resistance_scaling_fac
tor float]
[-min_unit_vertical_resistance_scaling_facto
r float]
[-max_unit_horizontal_capacitance_scaling_fa
ctor float]
[-max_unit_vertical_capacitance_scaling_fact
or float]
[-max_unit_horizontal_resistance_scaling_fac
tor float]
[-max_unit_vertical_resistance_scaling_facto
r float]
[-min_via_resistance float_resistance]
[-max_via_resistance float_resistance]
[-min_via_resistance_scaling_factor float]
[-max_via_resistance_scaling_factor float]
[-default]
```

set_design_license

Adds license information to the current design and can be used to require a license before a design can be read in.

```
status set_design_license
[-dont_show references]
[-quiet]
[-limited limited_keys]
regular_keys
```

set_design_top

Specifies the top-level design instance.

```
status set_design_top
instance_name
```

set_dft_clock_controller

Specifies clock controller parameters for DFT insertion.

```
int set_dft_clock_controller
[-cell_name controller_cell_name]
-design_name controller_design_name
[-ateclocks port_list]
[-pllclocks pin_list]
[-chain_count number_chains]
[-cycles_per_clock number_cycles]
[-test_mode_port port_name]
```

set_dft_configuration

Sets the DFT configuration for the current design.

```
int set_dft_configuration
[-scan enable | disable]
[-fix_clock enable | disable]
[-fix_set enable | disable]
[-fix_reset enable | disable]
[-fix_xpropagation enable | disable]
[-fix_bus enable | disable]
[-fix_bidirectional enable | disable]
[-control_points enable | disable]
[-observe_points enable | disable]
[-logicbist enable | disable]
[-wrapper enable | disable]
[-boundary enable | disable]
[-bsd enable | disable]
[-clock_controller enable | disable]
[-mode_decoding_style binary | one_hot]
[-scan_compression enable | disable]
[-pipeline_scan_data enable | disable]
[-connect_clock_gating enable | disable]
[-integration enable | disable]
```

set_dft_connect

Specifies a list of DFT connectivity associations. The specified source is connected to objects specified in the target list.

```
status set_dft_connect
string_label
-type clock_gating_control | scan_enable
-source port_or_pin
[-target object_list]
[-rise_target rise_clock_list]
[-fall_target fall_clock_list]
[-exclude cell_list]
```

set_dft_drc_configuration

Sets the DFT DRC configuration for the current design.

```
int set_dft_drc_configuration
[-internal_pins enable | disable]
[-pll_bypass enable | disable]
[-assume_pi_scan enable | disable]
[-assume_po_scan enable | disable]
[-static_x_analysis enable | disable]
[-use_test_model true | false]
[-use_test_mode core_mode_list]
[-clock_gating_init_cycles integer]
[-allow_se_set_reset_fix true | false]
```

set_dft_drc_rules

Sets the severity on the allowed set of violations.

```
int set_dft_drc_rules
[-error drc_error_ID]
[-warning drc_error_ID]
[-ignore drc_error_ID]
[-cell cell_list]
```

set_dft_equivalent_signals

Sets a given list of DFT signals as equivalents.

```
integer set_dft_equivalent_signals
signal_list
```

set_dft_insertion_configuration

Sets the DFT insertion configuration for the current design.

```
int set_dft_insertion_configuration
[-map_effort low | medium | high]
[-synthesis_optimization none | all]
[-route_scan_enable true | false]
[-route_scan_clock true | false]
[-route_scan_serial true | false]
[-preserve_design_name true | false]
[-unscan true | false]
```

set_dft_location

Specifies the DFT Hierarchy location for DFT insertion.

```
integer set_dft_location
dft_hier_name
```

set_dft_signal

Specifies the DFT signal types for DRC and DFT insertion.

```
status set_dft_signal
[-view existing_dft | spec]
[-test_mode mode_name]
-type signal_type
[-port port_list]
[-active_state active_state]
[-timing timing]
[-period period]
[-hookup_pin hookup_pin]
[-hookup_sense inverted | non_inverted]
[-internal_clocks none | single | multi]
[-ctrl_bits ctrl_bits_list]
[-pll_clock pll_clock]
[-ate_clock ate_clock]
[-differential_clock clock_port]
[-connect_to pin_list]
[-usage use_type]
[-associated_clock associated_clock]
[-associated_internal_clocks pin_names]
```

set_direct_power_rail_tie

Sets the direct_power_rail_tie attribute on library pins.

```
int set_direct_power_rail_tie
lib_pin_list [true | false]
```

set_disable_clock_gating_check

Disables the clock gating check for specified objects in the current design.

```
string set_disable_clock_gating_check
object_list
```

set_disable_timing

Disables timing arcs in the current design.

```
int set_disable_timing
object_list
[-from from_pin_name -to to_pin_name]
[-restore]
```

set_domain_supply_net

Set the primary power net and primary ground net of an already existing power_domain. This command is supported only in UPF mode.

```
int set_domain_supply_net
domain_name
-primary_power_net supply_net_name
-primary_ground_net supply_net_name
```

set_dont_retime

Sets the dont_retime attribute on cells and designs in the current design to prevent sequential cells from being moved by retiming optimizations.

```
int set_dont_retime
object_list
[true | false]
```

set_dont_touch

Sets the dont_touch attribute on cells, nets, references, and designs in the current design, and on library cells, to prevent modification or replacement of these objects during optimization.

```
status set_dont_touch
object_list
[true | false]
```

set_dont_touch_network

Sets the dont_touch_network attribute on clocks, pins, or ports in the current design to prevent cells and nets in the transitive fanout of the set_dont_touch_network objects from being modified or replaced during optimization.

```
status set_dont_touch_network
object_list
[-no_propagate]
```

set_dont_use

Sets the dont_use attribute on library cells to exclude them from the target library during optimization.

```
int set_dont_use
[-power] object_list
```

set_dp_int_round

Set the rounding positions on datapath output nets.

```
status set_dp_int_round
nets
external_rounding_position
[internal_rounding_position]
```

set_dp_smartgen_options

Controls the strategies that are used when datapath smart generation is active.

```
status set_dp_smartgen_options
[-all_options auto | true | false | default]
[-booth_encoding auto | true | false]
[-booth_radix8 auto | true | false]
[-booth_mux_based auto | true | false]
[-booth_cell auto | true | false]
[-mult_nand_based auto | true | false]
[-4to2_compressor_cell auto | true | false]
[-adder_radix auto | 2 | 3 | 4]
[-ling_adder auto | true | false]
[-hybrid_adder auto | true | false]
[-carry_select_adder_cell auto | true |
false]
[-cond_sum_adder auto | true | false]
[-bounded_fanout_adder auto | true | false]
[-mux_based auto | true | false]
[-inv_adder_cell auto | true | false]
[-sop2pos_transformation auto | true |
false]
[-tp_opt_tree auto | true | false]
[-tp_oper_sel auto | true | false]
[-smart_compare auto | true | false]
[-optimize_for default | area | speed |
area,speed]
[design or cell list]
```

set_drive

Sets the rise_drive or fall_drive attributes to specified resistance values on specified input and inout ports.

```
int set_drive
resistance
[-rise] [-fall] [-min] [-max]
port_list
```

set_driving_cell

Sets attributes on input or inout ports of the current design, specifying that a library cell or pin drives ports.

```
int set_driving_cell
[-lib_cell lib_cell_name]
[-library lib]
[-rise]
[-fall]
[-min]
[-max]
[-pin pin_name]
[-from_pin from_pin_name]
[-dont_scale]
[-no_design_rule]
[-none]
[-input_transition_rise rtran]
[-input_transition_fall ftran]
[-multiply_by factor]
port_list
[-cell
obsolete_-_please_use_-lib_cell_instead]
```

set_equal

Defines two input ports as logically equivalent.

```
int set_equal
port1
port2
```

set_extraction_options (Design Compiler topographical mode only)

Sets the parameters that influence extraction.

```
status set_extraction_options
[-max_process_scale max_process_scaling]
[-min_process_scale min_process_scaling]
[-default]
```

set_false_path

Removes timing constraints from particular paths.

```
int set_false_path
[-rise | -fall] [-setup | -hold]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
[-reset_path]
```

set_fanout_load

Sets the fanout_load attribute to a specified value on specified output ports of the current design.

```
int set_fanout_load
value
port_list
```

set_fix_hold

Sets a fix_hold attribute on clocks in the current design.

```
int set_fix_hold
clock_list
```

set_fix_multiple_port_nets

Sets the fix_multiple_port_nets attribute to a specified value on the current design or a list of designs.

```
int set_fix_multiple_port_nets
-default | -all
[-feedthroughs]
[-outputs]
[-constants]
[-buffer_constants]
[design_list]
```

set_flatten

Sets or removes the flatten attribute on specified designs or on the current design, to enable or disable the flattening optimization step during compile.

```
int set_flatten
[true | false]
[-effort low | medium | high]
[-minimize single_output | multiple_output |
none]
[-phase true | false] [-design design_list]
[-quiet]
```

set_fsm_encoding

Specifies the bit encodings for states in the current design.

```
int set_fsm_encoding
encoding_list
```

set_fsm_encoding_style

Defines the encoding style for assigning unencoded states.

```
int set_fsm_encoding_style
{one_hot | zero_one_hot | binary | gray |
auto | neutral}
```

set_fsm_minimize

Determines whether or not state minimization is to be performed on the state machine design during compile.

```
int set_fsm_minimize
true | false
```

set_fsm_order

Sets the ordering of states in a state machine design.

```
int set_fsm_order
state_list
```

set_fsm_preserve_state

Specifies states to be preserved during state minimization.

```
int set_fsm_preserve_state
state_list
```

set_fsm_state_vector

Specifies the instance names for flip-flops used to implement the state vector.

```
int set_fsm_state_vector
vector_list
```

set_fuzzy_query_options

Defines query options for fuzzy matching.

```
Boolean set_fuzzy_query_options
[-hierarchical_separators separator_list]
[-bus_name_notations bus_name_list]
[-class class_list]
[-regsub regsub]
[-regsub_cumulative]
[-reset]
[-show]
[-verbose]
```

set_host_options

Controls the maximum number of CPU cores that can be used for parallel execution.

```
status set_host_options
[-max_cores number_of_cores]
```

set_ideal_latency

Specifies ideal network latency.

```
string set_ideal_latency
[-rise | -fall]
[-min | -max]
delay
object_list
```

set_ideal_net

This command is replaced by set_ideal_network -no_propagate under the hood. It is recommended to use set_ideal_network instead of set_ideal_net.

```
status set_ideal_net
net_list
```

set_ideal_network

Marks a set of ports or pins in the current design as sources of an ideal network. This disables timing update and optimization of cells and nets in the transitive fanout of the specified objects.

```
integer set_ideal_network
object_list
[-dont_care_placement]
[-no_propagate]
```

set_ideal_transition

Specifies ideal transition for the ideal network & ideal nets.

```
string set_ideal_transition
[-rise | -fall]
[-min | -max]
transition_time
object_list
```

set_ignored_layers (Design Compiler topographical mode only)

Sets ignored routing layers for congestion analysis and RC estimation. This command is supported only in topographical mode.

```
int set_ignored_layers
[-rc_congestion_ignored_layers names]
[-min_routing_layer name]
[-max_routing_layer name]
```

set_impl_priority

Sets the formula attribute of the priority parameter and/or the set_id attribute for implementations in synthetic libraries.

```
status set_impl_priority
[-priority formula]
[-set_id id]
implementation_list
```

set_implementation

Specifies the implementation to use for synthetic library cell instances in a design.

```
status set_implementation
implementation_name
cell_list
[-check_impl]
```

set_input_delay

Sets input delay on pins or input ports relative to a clock signal.

```
status set_input_delay
delay_value
[-clock clock_name]
[-clock_fall]
[-level_sensitive]
[-network_latency_included]
[-source_latency_included]
[-rise]
[-fall]
[-max]
[-min]
[-add_delay]
port_pin_list
```

set_input_transition

Sets the max_transition_rise, max_transition_fall, min_transition_rise, or min_transition_fall attributes to the specified transition values on the specified input and inout ports.

```
int set_input_transition
transition
[-rise] [-fall] [-min] [-max]
port_list
```

set_isolate_ports

Specifies the ports that are to be isolated from internal fanouts of their driver nets.

```
int set_isolate_ports
[-type inverter | buffer]
[-driver cell_name]
[-force]
port_list
```

set_isolation

Defines the UPF isolation strategy for the power domains in the design. This command is supported only in UPF mode.

```
status set_isolation
isolation_strategy
-domain power_domain
-isolation_power_net isolation_power_net
-isolation_ground_net isolation_ground_net
[-clamp_value 0 | 1 | z | latch]
[-applies_to inputs | outputs | both]
[-elements objects]
[-no_isolation]
```

set_isolation_control

Provides additional options needed for creating isolation cells. This command is needed with most set_isolation commands. This command is supported only in UPF mode.

```
status set_isolation_control
isolation_strategy
-domain power_domain
-isolation_signal isolation_signal
[-isolation_sense low | high]
[-location self | parent]
```

set_leakage_power_model

Specifies the model that will be optimized by leakage optimizations.

```
status set_leakage_power_model
[-type leakage | channel_width]
[-mvth_weights weights]
[-reset]
```

set_level_shifter

Sets a strategy for level shifting during implementation. This command is supported only in UPF mode.

```
status set_level_shifter
level_shifter_name
-domain domain_name
[-elements list]
[-applies_to inputs | outputs | both]
[-threshold value]
[-rule low_to_high | high_to_low | both]
[-location self | parent | fanout |
automatic]
[-no_shift]
```

set_level_shifter_strategy

Sets the type of strategy to use for adjusting the voltage levels in the design.

```
int set_level_shifter_strategy
-rule all | low_to_high | high_to_low
[-location inside | outside | source | sink]
```

set_level_shifter_threshold

Sets the minimum threshold beyond which the voltage adjustment is required.

```
int set_level_shifter_threshold
-voltage volt
-percent diff
```

set_lib_attribute

Sets the value of an attribute on a library object.

```
list set_lib_attribute
object_list
attribute_name
attribute_value
```

set_libcell_dimensions

Sets the width and height of a library cell.

```
int set_libcell_dimensions
-cell cell_name
-width width
-height height
```

set_libpin_location

Sets the location of a pin of a library cell relative to the origin of the library cell.

```
int set_libpin_location
-cell library_cell_name
-pin pin_name_of_the_library_cell
-coordinate {x_coordinate y_coordinate}
```

set_load

Sets the load attribute to a specified value on specified ports and nets.

```
status set_load
value
objects
[-subtract_pin_load]
[-min]
[-max]
[[-pin_load] [-wire_load]]
```

set_local_link_library

Sets the local_link_library attribute to specified files and libraries on the current design.

```
int set_local_link_library
local_link_library
```

set_logic_dc

Specifies one or more input ports in the current design that are to be driven by don't care. The set_logic_one and set_logic_zero commands are used the same way as this command.

```
int set_logic_dc
port_list
```

set_logic_one

Specifies one or more input ports in the current design that are to be driven by logic one. The set_logic_zero and set_logic_dc commands are used the same way as this command.

```
int set_logic_one
port_list
```

set_logic_zero

Specifies one or more input ports in the current design that are to be driven by logic zero. The set_logic_one and set_logic_dc commands are used the same way as this command.

```
int set_logic_zero
port_list
```

set_logicbist_configuration

Controls the parameters for insertion (X)DBIST logic.

```
status set_logicbist_configuration
[-bist_ready true_or_false]
[-integration true_or_false]
[-codec_insertion true_or_false]
[-auto true_or_false]
[-type dbist_or_xdbist_or_sbist]
[-max_chain_length length]
[-codec_count n]
[-balance_bist_segments true_or_false]
[-invert_prpg_clock true_or_false]
[-prpg_shadow_si count]
[-prpg_length 257_or_479]
[-test_mode test_mode_names]
[-max_pattern_count max_pattern_count]
```

set_map_only

Sets the map_only attribute on specified objects so that they can be excluded from logic-level optimization during compile.

```
status set_map_only
object_list
[flag]
```

set_max_area

Sets the max_area attribute to a specified value on the current design.

```
int set_max_area
[-ignore_tns]
area_value
```

set_max_capacitance

Sets the max_capacitance attribute to a specified value on the specified input ports and designs.

```
int set_max_capacitance
capacitance_value
object_list
```

set_max_delay

Specifies a maximum delay target for paths in the current design.

```
int set_max_delay
delay_value
[-rise | -fall]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
[-group_path group_name]
[-reset_path]
```

set_max_dynamic_power

Sets the target dynamic power for the current design by setting the max_dynamic_power attribute to a specified value.

```
int set_max_dynamic_power
dynamic_power
[GW | MW | KW | W | mW | uW | nW | pW | fW |
aW]
```

set_max_fanout

Sets the max_fanout attribute to a specified value on specified input ports and/or designs.

```
int set_max_fanout
fanout_value
object_list
```

set_max_leakage_power

Sets the target leakage power for the current design by setting the max_leakage_power attribute to a specified value.

```
int set_max_leakage_power
leakage_power
[GW | MW | KW | W | mW | uW | nW | pW | fW |
aW]
```

set_max_lvth_percentage

Sets the the maximum percentage of the total cell area that can be of a low threshold voltage group.

```
status set_max_lvth_percentage
[max_lvth]
[-lvth_groups groups]
[-reset]
```

set_max_net_length

Sets the max_net_length attribute to a specified value on specified input ports and/or designs.

```
int set_max_net_length
net_length_value
object_list
```

set_max_time_borrow

Sets the max_time_borrow attribute to a specified value on clocks, latch cells, data pins, or clock (enable) pins, to constrain the amount of time borrowing possible for level-sensitive latches.

```
int set_max_time_borrow
delay_value
object_list
```

set_max_total_power

Sets the target total power for the current design by setting the max_total_power attribute to a specified value.

The set_max_total_power constraint will be obsolete in a future release. Use set_max_leakage_power and set_max_dynamic_power constraints as a replacement.

```
int set_max_total_power
total_power
[GW | MW | KW | W | mW | uW | nW | pW | fW |
aW]
```

set_max_transition

Sets the max_transition attribute to a specified value on specified clocks group, ports or designs.

```
int set_max_transition
transition_value
object_list
```

set_message_info

Set some information about diagnostic messages.

```
string set_message_info
-id message_id [-limit max_limit
 | -stop_on]
```

set_message_severity

Sets the severity level of the DFT message.

```
int set_message_severity
-names message_tags
severity
```

set_min_capacitance

Sets the min_capacitance attribute to a specified value on specified input ports in the current design.

```
int set_min_capacitance
capacitance_value
object_list
```

set_min_delay

Specifies a minimum delay target for paths in the current design.

```
int set_min_delay
delay_value
[-rise | -fall]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
[-reset_path]
```

set_min_library

Sets an alternate library to use for minimum delay analysis.

```
int set_min_library
max_library
-min_version min_library | -none
```

set_minimize_tree_delay

Sets the minimize_tree_delay attribute on a design or designs.

```
integer set_minimize_tree_delay
[true | false]
[-design ]
[design_list]
```

set_mode

Selects the mode of a component.

```
int set_mode
[mode_list]
[instance_list]
```

set_model_drive

Sets the model_drive attribute to a specified value on specified input or inout ports to set their drive values during synthetic library modeling.

```
int set_model_drive
drive_value
port_list
```

set_model_load

Sets the model_load attribute to a specified value on specified ports to set their load values during synthetic library modeling.

```
int set_model_load
load_value port_list
```

set_model_map_effort

Sets the model_map_effort attribute to a specified value on the current design, to specify the relative amount of CPU time to use during synthetic library modeling.

```
int set_model_map_effort
low | medium | high
```

set_multibit_options

Sets the multibit_mode and minimum_multibit_width attributes to specified values on the current design.

```
int set_multibit_options
[-default]
[-mode multibit_mode] [-minimum_width width]
```

set_multicycle_path

Modifies the single-cycle timing relationship of a constrained path.

```
integer set_multicycle_path
path_multiplier
[-rise | -fall]
[-setup | -hold]
[-start | -end]
[-from from_list
 | -rise_from rise_from_list
 | -fall_from fall_from_list]
[-through through_list]
[-rise_through rise_through_list]
[-fall_through fall_through_list]
[-to to_list
 | -rise_to rise_to_list
 | -fall_to fall_to_list]
[-reset_path]
```

set_mw_lib_reference

Sets the reference library for the Milkyway library.

```
status_value set_mw_lib_reference
[-mw_reference_library lib_list]
[-reference_control_file file_name]
libName
```

set_mw_technology_file

Sets the technology file of the Milkyway library.

```
status_value set_mw_technology_file
[-technology tech_file]
[-plib plib_file]
libName
```

set_nominal_power_state_template

Specifies a list of power nets as the template to define power states.

```
integer set_nominal_power_state_template
[power_net_list]
```

set_operand_isolation_cell

Specifies a list of operators or hierarchical combinational cells to be included or excluded as operand isolation candidates.

```
status set_operand_isolation_cell
object_list
[true | false]
```

set_operand_isolation_scope

Specifies whether a design or instance should be included or excluded for operand isolation processing.

```
status set_operand_isolation_scope
object_list
[true | false]
```

set_operand_isolation_slack

Sets the timing threshold to a value below which the automatic isolation roll back operation is not triggered.

```
status set_operand_isolation_slack
[slack_value]
[-weight weight_value]
```

set_operand_isolation_style

Sets the operand isolation style used by Power Compiler.

```
int set_operand_isolation_style
[-logic AND | OR | adaptive]
[-user_directives]
[-verbose]
```

set_operating_conditions

Defines the operating conditions for the current design.

```
int set_operating_conditions
[-analysis_type bc_wc | on_chip_variation]
[-min min_condition]
[-max max_condition]
[-min_library min_lib]
[-max_library max_lib]
[-min_phys min_proc]
[-max_phys max_proc]
[-library lib]
[-object_list objects]
[condition]
```

set_opposite

Defines two input ports as logically opposite.

```
int set_opposite
port1
port2
```

set_optimize_registers

Sets the optimize_registers attribute on the specified design or on the current design, so that compile automatically invokes the DC-Ultra optimize_registers command to retime the design during optimization.

```
status set_optimize_registers
[true | false]
[-minimum_period_only]
[-sync_transform multiclass | decompose |
dont_retime]
[-async_transform multiclass | decompose |
dont_retime]
[-check_design [-verbose]]
[-print_critical_loop]
[-clock clock_name [-edge rise | fall]]
[-latch]
[-justification_effort low | medium | high]
[-design ]
[design_list]
```

set_output_clock_port_type

Specifies the output clock port as a data port or a clock port.

```
status set_output_clock_port_type
-data | -clock
port_list
```

set_output_delay

Sets output delay on pins or output ports relative to a clock signal.

```
int set_output_delay
delay_value
[-clock clock_name [-clock_fall]
[-level_sensitive]]
[-network_latency_included]
[-source_latency_included]
[-rise]
[-fall]
[-max]
[-min]
[-add_delay]
[-group_path group_name]
port_pin_list
```

set_physical_hierarchy (Design Compiler topographical mode only)

Sets a list of cells to be physical hierarchies. This command is supported only in Design Compiler topographical mode.

```
status set_physical_hierarchy
object_list
```

set_pin_name_synonym

Defines synonyms for pin names.

```
Boolean set_pin_name_synonym
[-full_name]
[-force]
pin_name_synonym
pin_name
```

set_pipeline_scan_data_configuration

Specifies the pipeline scan data configuration for the design.

```
int set_pipeline_scan_data_configuration
[-head_pipeline_clock clock_name]
[-tail_pipeline_clock clock_name]
[-head_pipeline_stages integer]
[-head_pipeline_stages integer]
[-tail_pipeline_stages
Desired_number_of_tail_pipeline_stages]
```

set_placement_area (Design Compiler topographical mode only)

Creates the core placement area. This command is supported only in topographical mode.

```
int set_placement_area
-coordinate {X1 Y1 X2 Y2}
[-fixed | -unfixed]
```

set_port_fanout_number

Sets the number of external fanout points driven by specified ports in the current design.

```
status set_port_fanout_number
fanout_number
port_list
```

set_port_location

Annotates the specified top-level port with x- and y-coordinates and layer geometry, which the tool uses when running the reoptimize_design command.

```
status set_port_location
[-coordinate {x y}]
[-layer_name layer_name]
[-layer_area {lx ly ux uy}]
[-append]
port_name
```

set_port_side (Design Compiler topographical mode only)

Specifies the port side constraints for core generation. This command is supported only in topographical mode.

```
int set_port_side
port_list
-side {l | r | b | t | number}
```

set_power_gating_signal

Sets the attributes on the ports or pins in the design to indicate the power gating pins and the type of retention registers that will use the ports or pins to connect to or through the corresponding design hierarchy.

```
integer set_power_gating_signal
[-power_pin_index index]
[-library_pin library_pin_name]
[-type style]
ports_or_pins
```

set_power_gating_style

Sets the power_gating_style attributes on designs, cell instances, or HDL blocks, to specify the type of retention register cells from the target library used by the compile command.

```
int set_power_gating_style
-type style
[-hdl_blocks hdl_blocks]
[-hierarchy]
[cell_or_design_list]
```

set_power_prediction (Design Compiler topographical mode only)

Sets the power prediction mode for compile_ultra or compile_ultra -incremental. This command is supported only in topographical mode.

```
int set_power_prediction
[true | false]
[-ct_references lib_cell_list]
```

set_prefer

Sets the preferred attribute on specified library cells.

```
integer set_prefer
[-min]
cell_list
```

set_preferred_routing_direction (Design Compiler topographical mode only)

Sets the preferred routing direction for the specified routing layers.

```
string set_preferred_routing_direction
-layers list_of_layers
-direction horizontal | vertical
```

set_preferred_scenario

Sets the preferred scenario.

```
status set_preferred_scenario
[scenario_name]
```

set_propagated_clock

Specifies propagated clock latency.

```
string set_propagated_clock
object_list
```

set_pulse_clock_cell

Specifies the pulse type for a pulse clock cell with a single input and a single output.

```
string set_pulse_clock_cell
-type pulse_type
object_list
```

set_register_merging

Sets the register_merging attribute on the specified cells or designs, allowing register merging optimization on the objects.

```
status set_register_merging
obj_list
[true | false]
```

set_register_replication

Sets the register_replication attribute on the specified sequential cells, thus allowing register replication on the objects.

```
int set_register_replication
[-max_fanout max_fanout_value]
[-num_copies copy_value]
object_list
```

set_register_type

Sets the latch_type or flip_flop_type attributes on designs or cell instances, to specify which sequential cells from the target library are to be used by the compile command.

```
int set_register_type
-latch example_latch -exact |
-flip_flop example_flip_flop [-exact]
[cell_or_design_list]
```

set_related_supply_net

Associates an external supply net to the port of the design.

```
status set_related_supply_net
[supply_net_name]
[-object_list objects]
[-reset]
[-ground ground_net_name]
[-power power_net_name]
```

set_relative_always_on

Sets the relative always-on relationship between power domains.

```
status set_relative_always_on
domain_name
[-relative_to list_of_domains]
```

set_replace_clock_gates

Set directives for clock gate replacement. Forces the enabling or disabling of clock gate replacement for specified combinational cells in the current design. Also sets the edge type for modules or black-box cells that otherwise could not be replaced. The cells are replaced by executing the replace_clock_gates command.

```
int set_replace_clock_gates
[-include_cells cell_list]
[-exclude_cells cell_list]
[-rising_edge_clock pin_list]
[-falling_edge_clock pin_list]
[-undo object_list]
```

set_resistance

Sets the resistance value on nets.

```
int set_resistance
value
[-min] [-max] net_list
```

set_resource_allocation

Sets the resource_allocation attribute on the current design, specifying the type of resource allocation to be used by compile.

```
integer set_resource_allocation
none | area_only | area_no_tree_balancing |
constraint_driven
```

set_retention

Defines the UPF retention strategy for the power domains in the design. This command is supported only in UPF mode.

```
status set_retention
retention_strategy
-domain power_domain
-retention_power_net retention_power_net
-retention_ground_net retention_ground_net
[-elements objects]
```

set_retention_control

Defines the UPF retention control signals for the defined UPF retention strategy. This command is supported only in UPF mode.

```
status set_retention_control
retention_strategy
-domain power_domain
-save_signal {save_signal save_sense}
-restore_signal {restore_signal
restore_sense}
```

set_retention_control_pins

Converts the retention register library cell attributes in the old library format to the ones that can be used in $retain flow. The $retain flow requires retention register library cells to have new retention cell attributes, which is different from the original power gating flow.

```
status set_retention_control_pins
[-type style]
[-power_pin_index power_pin
 | -library_pin library_pin_name]
[-is_save_pin | -is_restore_pin |
-is_save_restore_pin]
lib_or_lib_cell_list
```

set_rp_group_options (Design Compiler topographical mode only)

Sets relative placement group attributes on the specified relative placement groups.

```
collection set_rp_group_options
rp_groups
[-alignment bottom-left | bottom-pin |
bottom-right]
[-pin_align_name pin_name]
[-utilization percentage]
[-ignore]
[-x_offset float]
[-y_offset float]
[-compress]
```

set_rtl_load

Sets an RTL load value for capacitance and resistance on pins, ports, and nets.

```
status set_rtl_load
[-min]
[-max]
pin_net_list
[-capacitance cvalue]
[-resistance rvalue]
```

set_scaling_lib_group

Specifies the scaling_lib_group to use for the current design, or a subdesign.

```
status set_scaling_lib_group
[-min min_group]
[-max max_group]
[-object_list objects]
[group]
```

set_scan_compression_configuration

Specifies the scan compression configuration for the design.

```
status set_scan_compression_configuration
[-minimum_compression compression_factor]
[-xtolerance high | default]
[-synchronize_chains none | all | tail |
head]
[-integration_only true | false]
[-hybrid true | false]
[-force_diagnosis true | false]
[-static_x_chain_isolation true | false]
[-chain_count scan_mode_chain_count]
[-max_length max_length_of_chains]
[-inputs number_of_inputs]
[-outputs number_of_outputs]
[-base_mode base_mode_name]
[-test_mode scan_compression_mode_name]
[-min_power true | false]
[-location compressor_decompressor_location]
```

set_scan_configuration

Specifies the scan chain design.

```
int set_scan_configuration
[-add_lockup true | false]
[-chain_count integer
 | -max_length integer_or_default
 | -exact_length integer_or_default
 | -count_per_domain integer_or_default]
[-clock_mixing no_mix | mix_edges |
mix_clocks | mix_clocks_not_edges]
[-create_dedicated_scan_out_ports true |
false]
[-internal_clocks single | none | multi]
[-insert_terminal_lockup true | false]
[-lockup_type latch | flip_flop]
[-mix_internal_clock_driver true | false]
[-partition vertical | horizontal]
[-style multiplexed_flip_flop | clocked_scan
| lssd | aux_clock_lssd | combinational |
none]
[-exclude_elements exclude_list]
[-voltage_mixing true | false]
[-power_domain_mixing true | false]
[-test_mode mode_name]
[-domain_based_scan_enable true | false]
[-pipeline_scan_enable true | false]
[-pipeline_fanout_limit integer_or_default]
[-reuse_mv_cells true | false]
[-create_test_clocks_by_system_clock_domain
true | false]
[-replace true | false]
[-hierarchical_isolation true | false]
[-shared_scan_in true | false]
[-preserve_multibit_segment true | false]
[-begin_and_end_with_rising_edge true |
false]
```

set_scan_element

Sets the scan_element attribute on specified design objects, to determine whether insert_dft replaces them with scan cells.

```
int set_scan_element
true | false
cell_design_ref_list
```

set_scan_group

Specifies an unordered group of cells that are not yet connected, but should be kept together within a scan chain. Also identifies existing logic in the current design that is to be designated as a scan segment.

```
status set_scan_group
scan_group_name
[-access signal_type_pin_pairs]
[-include_elements member_list]
[-serial_routed true | false]
[-segment_length length_of_virtual_segment]
[-class occ]
[-clock top_level_clock_port]
```

set_scan_link

Declares a scan link for the current design.

```
int set_scan_link
scan_link_name Wire | Lockup
[-test_mode mode_name]
```

set_scan_path

Specifies a scan chain for the current design.

```
integer set_scan_path
scan_chain_name
[-ordered_elements ordered_list]
[-head_elements head_list]
[-tail_elements tail_list]
[-include_elements include_list]
[-infer_dft_signals]
[-dedicated_scan_out true | false]
[-complete true | false]
[-exact_length length]
[-insert_terminal_lockup true | false]
[-scan_master_clock clock_name]
[-edge rising | falling]
[-scan_slave_clock clock_name]
[-scan_enable port_name]
[-scan_data_in port_name]
[-scan_data_out port_name]
[-test_mode mode_name]
[-view view_name]
[-class scan | wrapper | bsd]
[-hookup hookup_pin_list]
[-input_wrapper_cells_only enable | disable]
[-output_wrapper_cells_only enable |
disable]
[-pipeline_head_registers
scan_chain_element_names]
[-pipeline_tail_registers
scan_chain_element_names]
[-bsd_style global | synchronous |
asynchronous]
```

set_scan_register_type

Specifies a list of scan sequential cells from the target library that are to be used by insert_scan or compile -scan when scan replacing designs or cell instances.

```
int set_scan_register_type
[-exact]
[-type example_scan_seq_cell_list]
[cell_or_design_list]
```

set_scan_replacement

Specifies a table of one-to-one mappings of flip-flops to their equivalent scan flip-flops from the target library that are to be used by insert_dft when scan replacing cell instances.

Note that as of version Y-2006.06-SP1, compile -scan and compile_ultra -scan map to scan cells directly, rather than doing scan replacement. So, this command has no effect on compile -scan and compile_ultra -scan.

```
status set_scan_replacement
[-nonscan non_scan_seq_cell_list]
[-lssd lssd_rep_cell]
[-multiplexed_flip_flop muxed_scan_cell]
[-clocked_scan clocked_scan_cell]
[-aux_clock_lssd aux_clock_lssd]
[-combinational combinational_scan_cell]
```

set_scan_state

Sets the scan state status for the current db design.

```
int set_scan_state
unknown | test_ready | scan_existing
```

set_scan_suppress_toggling

Enables and controls the gating by the insert_dft command of functional output pins of scan flip-flops in order to suppress the toggling activity on the functional logic during scan shift.

```
status set_scan_suppress_toggling
-selection_method manual | auto | mixed
-include_elements cell_design_ref_list
-exclude_elements cell_design_ref_list
-min_slack minimum_timing_slack_after_gating
-ignore_timing_impact True | False
-total_percentage_gating percentage_value
```

set_scope

Specifies the current UPF scope. This command is supported only in UPF mode.

```
string set_scope
[instance]
```

set_size_only

Sets a list of attributes on specified leaf cells so that they can be sized only in optimization during compile.

```
int set_size_only
[-all_instances]
object_list
flag
```

set_structure

Sets structure attributes on a design or on a list of designs, to determine how the designs are structured during compile.

```
status set_structure
[true | false]
[-design design_list]
[-boolean true | false]
[-boolean_effort low | medium | high]
[-timing true | false]
```

set_svf

Generates a Formality setup information file for efficient compare point matching in Formality.

```
Boolean set_svf
filename
[-append]
[-off]
```

set_switching_activity

Sets switching activity annotation on nets, pins, ports and cells of the current design.

```
int set_switching_activity
[-static_probability sp_value]
[-toggle_rate tr_value]
[-state_dep state_condition]
[-path_dep path_sources]
[-rise_ratio ratio_value]
[-period period_value | -clock clock_name]
[-select select_types]
[-hier]
[-instances instances]
[object_list]
[-verbose]
```

set_synlib_dont_get_license

Specifies a list of synthetic library part licenses that are not automatically checked out.

```
int set_synlib_dont_get_license
license_list
```

set_tap_elements

Specifies the set of TAP state elements for the boundary-scan TAP controller.

```
int set_tap_elements
-state_cells cell_names
```

set_target_library_subset

Restricts optimization of a block to use a given subset of the target_library, for cases where operating condtion alone is not sufficient to define the subset.

```
int set_target_library_subset
[-object_list cells]
[-top]
library_list
[-milkyway_reflibs milkyway_reflib_paths]
```

set_test_assume

Sets the test_assume attribute to a logic value to be assumed on specified cell output pins throughout test design rule checking.

```
integer set_test_assume
zero_or_one_value
pin_list
```

set_test_point_element

Controls the configuration of the user-defined control, force, and observe test points.

```
int set_test_point_element
list_of_design_pin_objects
[-type control_0 | control_z0 | control_1 |
control_z1 | control_01 | control_z01 |
force_0 | force_z0 | force_1 | force_z1 |
force_01 | force_z01 | observe]
[-control_signal control_name]
[-clock_signal clock_name]
[-source_or_sink source_or_sink_name]
[-test_point_enable test_point_enable_name]
[-test_points_per_source_or_sink n]
[-test_points_per_test_point_enable n]
[-scan_source_or_sink enable | disable]
[-scan_test_point_enable enable | disable]
[-power_saving enable | disable]
```

set_testability_configuration

Controls configuration of automatically inserted control and observe points. The control and observe test point candidates are identified based on testability analysis.

```
int set_testability_configuration
[-type control | observe |
control_and_observe]
[-control_signal control_name]
[-clock_signal clock_name]
[-clock_type dominant | dedicated]
[-max_test_points n]
[-test_points_per_scan_cell n]
[-power_saving enable | disable]
[-max_additional_logic_area n]
```

set_timing_derate

Sets derate factors on the current design or specified objects. Derate factors specify upper and lower limits on delays for a particular operating condition.

```
int set_timing_derate
[-min]
[-max]
[-early]
[-late]
[-clock]
[-data]
[-net_delay]
[-cell_delay]
[-cell_check]
value
object_list
```

set_timing_ranges

Sets timing ranges for the current design.

```
int set_timing_ranges
[timing_ranges]
[-library library_name]
```

set_tlu_plus_files (Design Compiler topographical mode only)

Sets the files used for TLUPlus extraction. This command is supported only in topographical mode.

```
int set_tlu_plus_files
[-max_tluplus max_tluplus_string]
[-min_tluplus min_tluplus_string]
[-max_emulation_tluplus max_emul_string]
[-min_emulation_tluplus min_emul_string]
[-tech2itf_map mapping_file]
```

set_transform_for_retiming

Sets the transform_for_retiming attribute on cells in the current design. This can effect both hierarchical cells and sequential leaf cells.

```
integer set_transform_for_retiming
cell_list
multiclass | decompose | dont_retime
```

set_true_delay_case_analysis

Sets the true_delay_case_analysis attribute, which specifies the input vector value to use for specified pins or ports of the current design with the -true and -justify options of report_timing.

```
status set_true_delay_case_analysis
0 | 1 | r | f | none
port_pin_list
```

set_unconnected

Lists output ports to be unconnected.

```
int set_unconnected
port_list
```

set_ungroup

Sets the ungroup attribute on specified designs, cells, or references, indicating that they are to be ungrouped during compile.

```
int set_ungroup
object_list true | false
```

set_unix_variable

This is a synonym for the setenv command.

set_user_attribute

Sets the value of an attribute on a design or library object.

```
list set_user_attribute
object_list
attribute_name
attribute_value
[-bus]
[-quiet]
```

set_user_budget

Sets user budgets or budget ratios.

```
string set_user_budget
-from object_list
-to object_list
[-percent]
value
```

set_utilization (Design Compiler topographical mode only)

Specifies placement utilization constraint for core area. This command is supported only in topographical mode.

```
int set_utilization
util_float
```

set_voltage

Applies an operating voltage on a list of power net info objects.

```
int set_voltage
max_case_voltage
[-min min_case_value]
-object_list list_of_power_nets
```

set_vsdc

Generates a setup information file in V-SDC format for efficient compare point matching in formal verification tools.

```
Boolean set_vsdc
filename
[-append]
[-off]
```

set_wire_load_min_block_size

Sets the wire load min_block_size attribute on the current design.

```
int set_wire_load_min_block_size
size
```

set_wire_load_mode

Sets the wire_load_model_mode attribute on the current design, specifying how wire load models are to be used to calculate wire capacitance in nets.

```
status set_wire_load_mode
mode_name
```

set_wire_load_model

Sets the wire_load_attach_name attribute on designs, ports, hierarchical cells of current design, for selecting a wire load model to use in calculating wire capacitance.

```
status set_wire_load_model
-name model_name
[-library lib]
[-min] [-max]
[object_list]
```

set_wire_load_selection_group

Specify a selection group to use for determining a wire load model to be assigned to designs and cells or to a specified cluster. This command is supported only for the *enclosed* wire load mode.

```
int set_wire_load_selection_group
[-library lib] [-min]
[-max]
group_name [object_list]
```

set_wrapper_configuration

Sets the default wrapper configuration for the current design.

```
int set_wrapper_configuration
-class core_wrapper | shadow_wrapper
[-style dedicated | shared]
[-core core_list]
[-dedicated_cell_type WC_D1 | WC_D1_S |
none]
[-shared_cell_type WC_S1 | WC_S1_S | none]
[-dedicated_design_name design_name]
[-shared_design_name design_name]
[-register_io_implementation swap |
in_place]
[-use_dedicated_wrapper_clock true | false]
[-safe_state 0 | 1 | none]
[-delay_test true | false]
[-max_length integer]
[-chain_count integer]
[-mix_cells true | false]
[-input_shift_enable port_name]
[-output_shift_enable port_name]
[-maximize_reuse enable | disable]
[-reuse_threshold threshold_value]
[-use_system_clock_for_dedicated_wrp_cells
enable | disable]
```

set_zero_interconnect_delay_mode

Forces the timer to ignore the contribution on a timing path from any wire capacitance in the design.

```
status set_zero_interconnect_delay_mode
[true | false]
```

setenv

Sets the value of a system environment variable.

```
string setenv
variable_name new_value
```

sh

Executes a command in a child process.

```
string sh [args]
```

shell_is_in_topographical_mode

Determines if the Design Compiler shell was invoked in topographical mode.

```
status shell_is_in_topographical_mode
```

shell_is_in_upf_mode

Determines if the shell is in UPF mode.

```
status shell_is_in_upf_mode
```

shell_is_in_xg_mode

Determines if the shell is in XG mode.

```
status shell_is_in_xg_mode
```

simplify_constants

Propagates constants and other information in the current design.

```
status simplify_constants
[-boundary_optimization]
```

size_cell

Relinks leaf cells to a new library cell that has the required drive strength (or other properties).

```
collection size_cell
cell_object
lib_cell_object
```

sizeof_collection

Returns the number of objects in a collection.

```
int sizeof_collection
collection1
```

sort_collection

Sorts a collection based on one or more attributes, resulting in a new, sorted collection. The sort is ascending by default.

```
collection sort_collection
[-descending] collection1 criteria
```

source

Read a file and evaluate it as a Tcl script.

```
string source
[-echo] [-verbose] [-continue_on_error] file
```

start_gui

Starts the application GUI.

```
string start_gui
[-file name_of_script_file]
[-no_windows]
[-- x_args ...]
```

stop_gui

Stops the application GUI.

```
string stop_gui
```

sub_designs_of

Gets the subdesigns according to the options.

```
collection sub_designs_of
[-hierarchy]
[-in_partition | -partition_only]
[-dt_only | -ndt_only]
[-multiple_instances | -single_instances]
[-names_only]
design
```

sub_instances_of

Gets the subinstances according to the options.

```
collection sub_instances_of
[-hierarchy]
[-in_partition] [-partition_only]
[-dt_only] [-ndt_only]
[-of_references reference_list]
[-master_instance]
[-names_only]
design
```

suppress_message

Disables printing of one or more informational or warning messages.

```
string suppress_message
[message_list]
```

syntax_check

Enables or disables the Syntax Checker's syntax_check mode which checks commands for syntax errors.

```
integer syntax_check
true | false
```

translate

Translates a design from one technology to another.

```
int translate
[-preserve_structure]
```

```
int translate
[-preserve_structure]
```

unalias

Removes one or more aliases.

```
string unalias
pattern
```

ungroup

Removes a level of hierarchy.

```
status ungroup
cell_list | -all
[-prefix prefix_name]
[-flatten]
[-simple_names]
[-soft]
[-small n]
[-force]
[-start_level n]
[-all_instances]
```

uniquify

Removes multiply-instantiated hierarchy in the current design by creating a unique design for each cell instance.

```
int uniquify
[-force]
[-base_name base_name]
[-cell cell_list]
[-reference design_name]
[-new_name new_design_name]
[-dont_skip_empty_designs]
```

unsuppress_message

Enables printing of one or more suppressed informational or suppressed warning messages.

```
string unsuppress_message
[messages]
```

update_bounds (Design Compiler topographical mode only)

Updates an existing bound by adding or removing objects. The bound should be of type move bound.

```
int update_bounds
[-name bound_name]
[-add]
[-remove]
cell_list
```

update_lib

Reads in a specified library file and uses it to update an existing technology, synthetic, or symbol library.

```
int update_lib
[-overwrite] [-permanent] library_name
file_name [-no_warnings]
```

update_timing

Updates timing information on the current design.

```
int update_timing
```

use_test_model

Specifies the subdesigns that will use the Core Test Language (CTL) models during scan architect and insertion.

```
status use_test_model
[-true design_list]
[-false design_list]
```

which

Locates a file and displays its pathname.

```
string which
filename_list
```

while

Loop execution control structure.

win_select_objects (Design Compiler topographical mode only)

Creates a collection of objects equivalent to a graphical selection operation.

```
string win_select_objects
[-slct_targets slct_bus]
[-slct_targets_operation operation]
[-create_slct_buses]
[-root instance]
[ -within rectangle | -line line |-at point
|
 -radius r | -again_at ]
[-intersect]
[-index i]
```

win_set_filter (Design Compiler topographical mode only)

Sets a filter to apply to objects selected by the win_select_objects command.

```
int win_set_filter
-class class_name
[ -level level ]
[ -filter expression ]
[ -layer list ]
[-highlighted_only true|false ]
```

win_set_select_class (Design Compiler topographical mode only)

Sets the design objects to be collected by the win_select_objects command.

```
string win_set_select_class
{-all | class_names}
```

write

Writes a design netlist or schematic from memory to a file.

```
status write
[-format output_format]
[-hierarchy]
[-no_implicit]
[-modified]
[-output output_file_name]
[-names_file name_mapping_files]
[-donot_expand_dw]
[-scenarios scenario_list]
[design_list]
[-library library_name]
```

write_app_var

Writes a script to set the current variable values.

```
string write_app_var
-output file
[-all | -only_changed_vars]
[pattern]
```

write_bsdl

Generates the boundary-scan description language (BSDL) file for a boundary-scan design.

```
int write_bsdl
[-naming_check VHDL | BSDL | none]
[-output file_name]
[-effort low | medium | high]
```

write_compile_script

Writes a compile script for the specified design.

```
int write_compile_script
[-absolute_paths]
[-hierarchy]
[-no_reports]
[-refining]
-destination pass
[-update design_list]
[design]
```

write_design_lib_paths

Writes into a file the paths to which design libraries are mapped.

```
status write_design_lib_paths
[-filename file_name]
[-dc_setup]
```

write_environment

Writes the variable settings and constraints for the specified cells or designs.

```
return_val write_environment
[-cells cell_list | -designs design_list]
[-format dcsh | dctcl]
[-output file_name]
[-suffix suffix]
[-environment_only]
[-constraints_only]
[-no_lib_info]
[-consistency]
```

write_file

Writes a design netlist or schematic from memory to a file.

```
int write_file
[-format output_format]
[-hierarchy]
[-no_implicit]
[-modified]
[-output output_file_name]
[-names_file name_mapping_files]
[-donot_expand_dw]
[-scenarios scenario_list]
[design_list]
[-library library_name]
```

write_interface_timing

Generates an interface timing ASCII report for a gate-level netlist or an interface logic model (ILM).

```
int write_interface_timing
file_name
[-ignore_ports port_list]
[-significant_digit digits]
[-nosplit]
```

write_lib

Writes a compiled library to disk in Synopsys database, EDIF, or VHDL format.

```
int write_lib
library_name
[-format db | edif | vhdl]
[-compress compression_format]
[-output file_name]
[-names_file file_list]
[-macro_only]
```

write_link_library

Writes shell commands to save the current link library settings for design instances.

```
int write_link_library
[-full_path_lib_names] [-nosplit]
[-full_path_lib_names] [-nosplit]
[-output file_name]
[-target target]
```

write_makefile

Writes a makefile that defines the dependencies and commands required to compile the specified design.

```
int write_makefile
[-absolute_paths]
[-dependencies depends]
-destination pass
[-target target_name]
[-lsf [-bsubargs bsub_args]]
[-update design_list]
[design]
```

write_milkyway

Writes out the design to Milkyway database format.

```
status write_milkyway
-output filename
[-overwrite]
[-scenario scenario_list]
```

write_mw_lib_files

Writes the technology, or plib, or reference control file of the Milkyway library.

```
status_value write_mw_lib_files
[-technology]
[-plib]
[-reference_control_file]
-output file_name
libName
```

write_parasitics

Writes parasitics in SPEF format or as a Tcl script containing set_load and set_resistance commands.

```
status write_parasitics
[-output file_name]
[-format reduced | distributed]
[-min]
[-ratio ratio_number]
[-script]
```

write_partition

Writes the database for a design into the Automated Chip Synthesis data structure.

```
int write_partition
-type pre | post
[-destination pass]
[-hierarchy]
[design]
[-format db | ddc]
```

write_partition_constraints

Writes out the timing constraints for a design.

```
int write_partition_constraints
[-hierarchy]
-destination pass
[-update design_list]
[design]
```

write_physical_constraints (Design Compiler topographical mode only)

Writes the script of Tcl commands to export the current physical constraint settings. This command is supported only in topographical mode.

```
status write_physical_constraints
-output tcl_file
[-site_row]
[-pre_route]
```

write_rp_groups (Design Compiler topographical mode only)

Writes out the relative placement constraints for the specified relative placement groups.

```
collection write_rp_groups
rp_groups | -all
[-hierarchy]
[-quiet]
[-nosplit]
[-output filename]
[-create]
[-leaf]
[-keepout]
[-instance]
[-include]
```

write_rtl_load

Writes a script of RTL load commands for the current design.

```
int write_rtl_load
[-format dctcl | dcsh]
[-output file_name]
```

write_saif

Writes a backward Switching Activity Interchange Format (SAIF) file.

```
int write_saif
-output file_name
[-instances instances]
[-no_hier]
[-rtl]
[-propagated]
[-exclude_sdpd]
```

write_scan_def

Writes scan chain information in SCANDEF format for performing scan chain reordering using place and route tools.

```
int write_scan_def
[-output output_command_file]
[-expand_elements
List_of_design_instance_names_that_should_be
_expanded]
```

write_script

Writes shell commands to save the current settings.

```
int write_script
[-hierarchy]
[-no_annotated_check] [-no_annotated_delay]
[-no_cg]
[-full_path_lib_names] [-nosplit]
[-format dctcl | dcsh]
[-include loop_breaking]
[-output file_name]
```

write_sdc

Writes out a script in Synopsys Design Constraints (SDC) format.

```
int write_sdc
file_name
[-nosplit]
[-version sdc_version]
```

write_sdf

Writes a Standard Delay Format (SDF) back-annotation file.

```
string write_sdf
[-version sdf_version] [-significant_digits
digits]
[-instance inst_name]
file_name
```

write_test

Formats the test patterns for the current design into one or more test vector files.

```
status write_test
[-output output_vector_file_name]
[-format stil | stil_testbench | wgl_serial
| verilog]
```

write_test_model

Writes a test model file.

```
int write_test_model
[-format ctl | ddc]
[-names verilog | verilog_single_bit]
[-output model_file]
[-inclusive]
[-design design_name]
```

write_test_protocol

Writes a test protocol file.

```
int write_test_protocol
[-design design_name]
[-output file_name]
[-test_mode mode_name]
[-names format_name]
[-pll_bypass]
[-instruction instruction_name]
```

Synthesis Variables

Synthesis tools define variables that are used to control the behavior of the tools.

For a list of all variables and their current values, enter the following command from within dc_shell:

```
dc_shell> printvar
```

The syntax for setting a variable is

```
dc_shell> set variable_name value
```

access_internal_pins

Controls the creation, deletion, and user access of internal pins.

Default value for this variable is false.

acs_area_report_suffix

Specifies the suffix for area reports generated during the automated compile process.

Default value for this variable is area.

acs_autopart_max_area

Defines partition threshold; used with other acs variables to control chip-level partitioning.

Default value for this variable is 0.0.

acs_autopart_max_percent

Controls chip-level partitioning; used with other acs variables.

Default value for this variable is 0.0.

acs_budgeted_cstr_suffix

Specifies the suffix for constraint files generated by the derive_partition_budgets command.

Default value for this variable is con.

acs_compile_script_suffix

Specifies the default suffix for script files generated by the write_compile_script command, sourced in the makefile generated by the write_makefile command, and located by the report_pass_data command.

Default value for this variable is autoscr.

acs_constraint_file_suffix

Specifies the default suffix for constraint files generated by write_partition_constraints during the automated compile process.

Default value for this variable is con.

acs_cstr_report_suffix (not supported in Design Compiler topographical mode)

Specifies the default suffix for constraint reports generated during the automated compile process. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is cstr.

acs_db_suffix (not supported in Design Compiler topographical mode)

Specifies the default suffix for .db files that are read or written during the automated compile process. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is db.

acs_dc_exec

Specifies the location of the dc_shell executable. This variable is used by the acs_compile_design, acs_refine_design, and acs_recompile_design commands to generate the makefile.

Default value for this variable is "".

acs_default_pass_name

Specifies the prefix for the default data directory names. The pass number (either 0 or 1) is added to the prefix to generate the directory name.

Default value for this variable is pass.

acs_dir

This variable maps ACS directories to file types. This variable is used by the acs_compile_design, acs_refine_design, and acs_recompile_design commands to generate the ACS directory structure.

Default value for this variable is .

acs_exclude_extensions

Specifies the file endings of files you do not want the acs_read_hdl command to analyze.

Default value for this variable is "".

acs_exclude_list

Specifies files and directories you do not want the acs_read_hdl command to analyze.

Default value for this variable is "[list $synopsys_root]".

acs_global_user_compile_strategy_script (not supported in Design Compiler topographical mode)

Specifies the base file name for the user-defined default compile strategy. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is default.

acs_hdl_source

Specifies the location of the source code files analyzed by the acs_read_hdl command.

Default value for this variable is "".

acs_hdl_verilog_define_list

Specifies a list of text macros which are to be defined by the acs_read_hdl command during verilog file analysis.

Default value for this variable is "".

acs_lic_wait

Specifies the maximum wait time for checking out all the licenses required by a compile job.

Default value for this variable is 0.

acs_log_file_suffix (not supported in Design Compiler topographical mode)

Specifies the default suffix for log files generated during the automated compile process. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is log.

acs_make_args

Specifies the command arguments for the make utility command (gmake, by default).

Default value for this variable is set acs_make_args.

acs_make_exec

Specifies the make utility command used to run the compile jobs.

Default value for this variable is gmake.

acs_makefile_name (not supported in Design Compiler topographical mode)

Specifies the file name for the makefile generated by the write_makefile command and run by the compile_partitions command. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is makefile.

acs_num_parallel_jobs

Specifies the number of compile jobs to run in parallel when using gmake as the make utility.

Default value for this variable is 1.

acs_override_report_suffix

Specifies the suffix for user-defined partition report scripts.

Default value for this variable is report.

acs_override_script_suffix (not supported in Design Compiler topographical mode)

Specifies the suffix for user-defined partition compile scripts. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is scr.

acs_qor_report_suffix (not supported in Design Compiler topographical mode)

Specifies the suffix for QOR reports generated during the automated compile process; the default is *qor*. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is qor.

acs_svf_suffix

Specifies the default suffix for svf files that are written during the automated compile process.

Default value for this variable is svf.

acs_timing_report_suffix (not supported in Design Compiler topographical mode)

Specifies the suffix for timing reports generated during the automated compile process. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is tim.

acs_use_autopartition

Sets autopartitioning as the default partitioning strategy for the chip-level compile commands.

Default value for this variable is false.

acs_use_default_delays

Sets top-down environment propagation as the constraint generation method for GTECH designs (the acs_compile_design command). When this variable is false (the default), the acs_compile_design command uses RTL budgeting to generate constraints for the compile partitions.

Default value for this variable is false.

acs_user_budgeting_script

Specifies the file name for the user-defined budgeting script.

Default value for this variable is budget.scr.

acs_user_compile_strategy_script_suffix (not supported in Design Compiler topographical mode)

Specifies the suffix for user-defined partition compile strategies. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is compile.

acs_verilog_extensions

Specifies the file endings of files analyzed as Verilog source code by the acs_read_hdl command.

Default value for this variable is ".v".

acs_vhdl_extensions

Specifies the file endings of files the acs_read_hdl command analyzes as VHDL source code.

Default value for this variable is ".vhd".

acs_work_dir (not supported in Design Compiler topographical mode)

Specifies the root of the Automated Chip Synthesis project directory. For use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is the current working directory.

alib_library_analysis_path

Specifies a single path, similar to a search path, for reading and writing the alib files that correspond to the target libraries.

Default value for this variable is "./".

auto_insert_level_shifters

Setting this variable to false would prevent automatic level shifter insertion in commands like compile, insert_dft etc

Default value for this variable is true.

auto_insert_level_shifters_on_clocks

This variable is used to direct automatic level shifter insertion to insert level shifter on specified clocks

Default value for this variable is "".

auto_link_disable

Specifies whether the code to perform an auto_link during any Design Compiler command should be disabled.

Default value for this variable is false.

auto_link_options

Specifies the link command options to be used when link is invoked automatically by various Design Compiler and DFT Compiler commands (for example, create_schematic and compile).

Default value for this variable is -all.

auto_ungroup_preserve_constraints

Preserves (when *true*) the timing constraints on the hierarchy when the hierarchy is ungrouped during the process of optimization.

Default value for this variable is true.

auto_wire_load_selection (not supported in Design Compiler topographical mode)

Controls automatic selection of wire load model.

Default value for this variable is area_locked.

bind_unused_hierarchical_pins

Specifies if unused input and output hierarchical pins should be connected to constant tie-off cells during compile.

Default value for this variable is true.

bsd_max_in_switching_limit

Specifies the maximum number of design inputs that may switch simultaneously while generating input DC parametric tests using the create_bsd_patterns command.

Default value for this variable is 60000.

bsd_max_out_switching_limit

Specifies the maximum number of design outputs that may switch simultaneously while generating output DC parametric tests using the create_bsd_patterns command.

Default value for this variable is 60000.

bsd_physical_effort

Default value for this variable is medium.

budget_generate_critical_range

Enables automatic generation of set_critical_range commands by dc_allocate_budgets for multiply-instantiated subdesigns.

Default value for this variable is false.

budget_map_clock_gating_cells

Maps integrated clock gating cells into target library during RTL budgeting.

Default value for this variable is false.

bus_inference_descending_sort

Specifies that the members of that port bus are to be sorted in descending order rather than in ascending order.

Default value for this variable is true.

bus_inference_style

Specifies the pattern used to infer individual bits into a port bus.

Default value for this variable is "".

bus_minus_style

Controls the naming of individual members of bit-blasted port, instance, or net buses with negative indices.

Default value for this variable is -%d.

bus_multiple_separator_style

Determines the name of a multibit cell that implements bits that do not form a range.

Default value for this variable is ,.

bus_naming_style

Specifies the style to use in naming an individual port member, net member, or cell instance member of an EDIF array or of a Verilog or VHDL vector.

Default value for this variable is %s[%d].

bus_range_separator_style

Specifies the style to use in naming a net connected to the "wire" end of a ripper in the EDIF file.

Default value for this variable is :.

cache_dir_chmod_octal

Specifies the value of the mode bits for created cache directories.

Default value for this variable is 777.

cache_file_chmod_octal

Specifies the value of the mode bits for created cache files.

Default value for this variable is 666.

cache_read

Specifies a list of directories that contain cache files that will be read from whenever a cache entry is needed.

Default value for this variable is ~.

cache_read_info

Specifies whether an informational message will be printed each time a cache element is read.

Default value for this variable is false.

cache_write

Specifies the directory where optimized and unoptimized synlib parts will be written, if they are not already in the cache.

Default value for this variable is ~.

cache_write_info

Specifies whether an informational message will be printed each time a cache element is written.

Default value for this variable is false.

case_analysis_log_file

Specifies the name of a log file generated during propagation of constant values, from case analysis or from nets tied to logic zero or logic one. Each scenario has its proprietary log file if multiple scenarios exist.

Default value for this variable is "".

case_analysis_propagate_through_icg

Determines whether case analysis is propagated through integrated clock gating cells.

Default value for this variable is false.

case_analysis_with_logic_constants

When true, enables constant propagation, even if a design contains only logic constants.

Default value for this variable is false.

change_names_bit_blast_negative_index

Bit blast the bus if any bit of it is negative.

Default value for this variable is false.

change_names_dont_change_bus_members

Controls how the change_names command modifies the names of bus members.

Default value for this variable is false.

check_design_allow_non_tri_drivers_on_tri_bus

Specifies the severity level to be applied during compile or check_design command execution, when three-state buses with non three-state driver(s) are found in the design.

Default value for this variable is true.

check_design_allow_unknown_wired_logic_type

Specifies the severity level to be applied during compile or check_design command execution, when nets with multiple drivers(unknown wired-logic type) are found in the design.

Default value for this variable is true.

check_error_list

Specifies the error codes that the check_error command checks for.

Default value for this variable is CMD-004 CMD-006 CMD-007 CMD-008 CMD-009 CMD-010 CMD-011 CMD-012 CMD-014 CMD-015 CMD-016 CMD-019 CMD-026 CMD-031 CMD-037 DB-1 DCSH-11 DES-001 ACS-193 FILE-1 FILE-2 FILE-3 FILE-4 LINK-7 LINT-7 LINT-20 LNK-023 OPT-100 OPT-101 OPT-102 OPT-114 OPT-124 OPT-127 OPT-128 OPT-155 OPT-157 OPT-181 OPT-462 UI-11 UI-14 UI-15 UI-16 UI-17 UI-19 UI-20 UI-21 UI-22 UI-23 UI-40 UI-41 UID-4 UID-6 UID-7 UID-8 UID-9 UID-13 UID-14 UID-15 UID-19 UID-20 UID-25 UID-27 UID-28 UID-29 UID-30 UID-32 UID-58 UID-87 UID-103 UID-109 UID-270 UID-272 UID-403 UID-440 UID-444 UIO-2 UIO-3 UIO-4 UIO-25 UIO-65 UIO-66 UIO-75 UIO-94 UIO-95 EQN-6 EQN-11 EQN-15 EQN-16 EQN-18 EQN-20.

collection_result_display_limit

Sets the maximum number of objects that can be displayed by any command that displays a collection.

Default value for this variable is 100.

command_log_file

Specifies the name of the file to which a log of the initial values of variables and commands executed is written. If the value is an empty string, a command log file is not created.

Default value for this variable is "./command.log".

company

Specifies the name of the company where Synopsys software is installed. The company name is displayed on the schematics.

Default value for this variable is "".

compatibility_version

Sets the default behavior of the system to be the same as the Synopsys software version specified in the variable.

Default value for this variable is C-2009.06.

compile_allow_dw_hierarchical_inverter_opt

Allows the phase inversion boundary optimization to be applied to DesignWare components.

Default value for this variable is false.

compile_assume_fully_decoded_three_state_busses

Specifies whether the compile and translate commands can assume that three-state busses are fully decoded.

Default value for this variable is false.

compile_auto_ungroup_area_num_cells

Defines the minimum number of child cells a design hierarchy must have so that the command compile -auto_ungroup area does not ungroup the hierarchy.

Default value for this variable is 30.

compile_auto_ungroup_count_leaf_cells

Determines if the number of leaf cells should be counted or the number of child cells in the immediate hierarchy should be counted to compare against the value of variable compile_auto_ungroup_area_num_cells in area-based auto-ungroup

Default value for this variable is false.

compile_auto_ungroup_override_wlm

Specifies whether the compiler considers a cell instance for automatic ungrouping, if the cell's wire load model differs from that of its parent.

Default value for this variable is false.

compile_automatic_clock_phase_inference

Specifies the method used to determine clock phase during sequential mapping.

Default value for this variable is strict.

compile_checkpoint_phases (not supported in Design Compiler topographical mode)

Determines whether checkpoints are generated during execution of the compile command.

Default value for this variable is false.

compile_clock_gating_through_hierarchy

Controls whether the compile or compile_ultra command with the -gate_clock option will perform clock gating through hierarchy boundaries.

Default value for this variable is false.

compile_cpu_limit (not supported in Design Compiler topographical mode)

Specifies a time, in minutes, to be used as the limit for the amount of time to be spent in the phases after structuring and mapping. Optimization cancels when the limit is reached.

Default value for this variable is 0.0.

compile_create_wire_load_table (not supported in Design Compiler topographical mode)

Controls the type of wire load model generated by the create_wire_load command.

Default value for this variable is false.

compile_delete_unloaded_sequential_cells

Controls whether the compile/physopt/compile_physical command deletes unloaded sequential cells.

Default value for this variable is true.

compile_disable_hierarchical_inverter_opt

Controls whether inverters can be moved across hierarchical boundaries during boundary optimization.

Default value for this variable is false.

compile_dont_touch_annotated_cell_during_inplace_opt (not supported in Design Compiler topographical mode)

Controls whether cells that have annotated delays can be optimized.

Default value for this variable is false.

compile_dont_use_dedicated_scanout

Controls whether optimizations use a scan cell's dedicated scan-out pin for functional connections.

Default value for this variable is 1.

compile_enable_dyn_max_cap

Determines whether the tool is to use the operating frequency to determine the maximal pin load.

Default value for this variable is false.

compile_fix_cell_degradation (not supported in Design Compiler topographical mode)

Controls whether the algorithms for fixing cell degradation violation are activated.

Default value for this variable is false.

compile_hold_reduce_cell_count (not supported in Design Compiler topographical mode)

Controls whether the logic used to fix hold time violations is selected based on minimum cell count or minimum area.

Default value for this variable is false.

compile_implementation_selection

Controls whether the compile command reevaluates the current implementation of a synthetic module during optimization.

Default value for this variable is true.

compile_instance_name_prefix

Specifies the prefix used in generating cell instance names when compile is executed.

Default value for this variable is U.

compile_instance_name_suffix

Specifies the suffix used for generating cell instance names when compile is executed.

Default value for this variable is "".

compile_keep_original_for_external_references

Controls compile command to keep the original design when there is an external reference to the design.

Default value for this variable is false.

compile_log_format

Controls the format of the columns to be displayed during the mapping phases of compile and reoptimize_design.

Default value for this variable is %elap_time %area %wns %tns %drc %endpoint.

compile_negative_logic_methodology

Specifies the logic value connected to floating inputs by the compile and translate commands.

Default value for this variable is false.

compile_no_new_cells_at_top_level

Controls whether the compile command adds new cells to the top-level design.

Default value for this variable is false.

compile_power_domain_boundary_optimization

Sets the variable to false to disable boundary optimization across power domain boundaries.

Default value for this variable is true.

compile_preserve_subdesign_interfaces

Controls whether the compile command preserves the subdesign interface.

Default value for this variable is false.

compile_retime_exception_registers

Controls whether registers with common path exceptions, including max_path, min_path, multicycle_path, false_path, and group_path, can be moved by adaptive retiming.

Default value for this variable is false.

compile_retime_license_behavior

Controls how the compile command behaves when the optimize_registers or balance_registers attributes are set on a design or parts of a design and the required license(s) (BOA-BRT or DC-Expert) are not available immediately.

Default value for this variable is wait.

compile_seqmap_enable_output_inversion

Controls whether the compile command allows sequential elements to have their output phase inverted. Has no effect on the compile_ultra command.

Default value for this variable is false.

compile_seqmap_identify_shift_registers

Controls the identification of shift registers in compile -scan. This feature is only supported in test-ready compile with Design Compiler Ultra with a multiplexed scan-style.

Default value for this variable is true.

compile_seqmap_identify_shift_registers_with_synchronous_logic

Controls whether shift registers that contain synchronous logic between the registers are identified. This variable only has an effect in Design Compiler Ultra optimization when shift-register identification is enabled with the compile_seqmap_identify_shift_registers variable.

Default value for this variable is true.

compile_seqmap_propagate_constants

Controls whether the compile command tries to identify and remove constant registers and propagate the constant value throughout the design.

Default value for this variable is true.

compile_seqmap_propagate_high_effort

Controls whether the compile command considers registers that cannot escape their reset state to be constant.

Default value for this variable is true.

compile_slack_driven_buffering

Controls whether rule based optimization will run additional steps of slack-driven buffering in ultra mode.

Default value for this variable is false.

compile_top_acs_partition

Controls whether the compile -top command only fixes all violations that cross top-level partitions.

Default value for this variable is false.

compile_top_all_paths

Controls whether the compile -top command fixes all violations in the design, or only those that cross top-level hierarchical boundaries.

Default value for this variable is false.

compile_ultra_ungroup_dw

Determines whether to unconditionally ungroup DesignWare cells in compile_ultra flow.

Default value for this variable is true.

compile_ultra_ungroup_small_hierarchies

Determines whether to automatically ungroup small hierarchies in the compile_ultra flow.

Default value for this variable is true.

compile_update_annotated_delays_during_inplace_opt (not supported in Design Compiler topographical mode)

Controls whether compile -in_place can update annotated delay values in the neighborhood of swapped cells. It has no effect for reoptimize_design and physopt, which always update annotated delay values.

Default value for this variable is true.

compile_use_fast_delay_mode

Selects the algorithm used for delay calculations when using the CMOS2 or nonlinear delay models.

Default value for this variable is true.

complete_mixed_mode_extraction (Design Compiler topographical mode only)

Enables extraction of both routed and unrouted nets with a single command. This is known as mixed-mode extraction.

Default value for this variable is true.

context_check_status

Reports whether the context_check mode is enabled (read-only).

Default value for this variable is false.

create_clock_no_input_delay (not supported in Design Compiler topographical mode)

Affects delay propagation characteristics of clock sources created by using the create_clock command.

Note: This variable will become obsolete. Please adjust your scripts accordingly.

Default value for this variable is false.

current_design

Specifies the design being worked on. This variable is used by most of the Synopsys commands.

Default value for this variable is "".

db_load_ccs_data

The variable is obsolete and is not needed any more. The tool takes care of loading the CCS timing information automatically.

Default value for this variable is false.

dc_shell_mode

Reports the mode of the current dc_shell session.

Default value for this variable is tcl.

dct_placement_ignore_scan

Sets the flag for the placer to ignore scan connections in Design Compiler topograpical.

Default value for this variable is false.

dct_prioritize_area_correlation (Design Compiler topographical mode only)

Determines if optimization should prioritize area correlation between DC-T and ICC.

Default value for this variable is false.

ddc_allow_unknown_packed_commands

Causes the read_file command to attempt to read DDC files which contain packed commands which are unknown to the current version of dc_shell.

Default value for this variable is true.

default_input_delay

Specifies the global default input delay value to be used for environment propagation.

Default value for this variable is 30.

default_name_rules

Contains the name of a name rule to be used as a default by the change_names command, if the command's -rules option does not specify a *name_rules* value.

Default value for this variable is "".

default_output_delay

Specifies the global default output delay value to be used for environment propagation.

Default value for this variable is 30.

default_port_connection_class

Contains the value of the connection class to be assigned to ports that do not have a connection class assigned to them.

Default value for this variable is universal.

default_schematic_options

Specifies options to use when schematics are generated.

Default value for this variable is -size infinite.

designer

Specifies the name of the current user.

Default value for this variable is "".

disable_auto_time_borrow

Determines whether the report_timing command and other commands will use automatic time borrowing.

Default value for this variable is false.

disable_case_analysis

When true, disables constant propagation from both logic constants and set_case_analysis command constants.

Default value for this variable is false.

disable_library_transition_degradation

Controls whether the transition degradation table is used to determine the net transition time.

Default value for this variable is false.

disable_mdb_stop_points

Disable 'stop at any level' functionality in write_mdb command.

Default value for this variable is false.

do_operand_isolation

Enables or disables operand isolation as a dynamic power optimization technique for a design.

Default value for this variable is false.

dont_touch_nets_with_size_only_cells

Specifies whether a net is marked dont touch if it is driven by at least one cell marked size_only and drives at least one cell marked by size_only.

Default value for this variable is false.

dpcm_arc_sense_mapping

Controls whether Design Compiler maps half unate arcs to preset and clear arcs for sequential cells.

Default value for this variable is true.

dpcm_debuglevel

Determines the level of debugging for Design Compiler.

Default value for this variable is 0.

dpcm_functionscope

Controls how DPCM determines the FunctionalMode value when doing timing calculations.

Default value for this variable is global.

dpcm_level

Controls the mode used by DPCM for timing calculations.

Default value for this variable is performance.

dpcm_libraries

Specifies the libraries of the link_path that use DPCM delay calculation.

Default value for this variable is "".

dpcm_rulepath

Specifies a list of paths, similar to a search path, for locating the DPCM main rule.

Default value for this variable is "".

dpcm_rulespath

Locates the DPCM subrules when provided a list of paths, the list being similar to a search path.

Default value for this variable is "".

dpcm_slewlimit

Determines Design Compiler behavior when input pin slew exceeds library limits.

Default value for this variable is true.

dpcm_tablepath

Specifies a list of paths, similar to a search path, for locating the DPCM tables.

Default value for this variable is "".

dpcm_temperaturescope

Controls whether DPCM requests Temperature for each delay calculation.

Default value for this variable is global.

dpcm_version

Specifies the version of the API used by the DPCM delay calculation.

Default value for this variable is IEEE-P1481.

dpcm_voltagescope

Controls whether DPCM requests RailVoltage values for every delay calculation.

Default value for this variable is global.

dpcm_wireloadscope

Controls whether DPCM requests WireLoadModel values for every delay calculation.

Default value for this variable is global.

duplicate_ports

Specifies whether ports are to be drawn on every sheet for which an input or output signal appears.

Default value for this variable is false.

echo_include_commands

Controls whether the contents of a script file is printed as it executes.

Default value for this variable is true.

enable_cell_based_verilog_reader

This variable turns on the verilog2cel verilog reader.

Default value for this variable is false.

enable_instances_in_report_net

Enables report_net to report on instances in the current design.

Default value for this variable is true.

enable_page_mode

Controls whether long reports are displayed one page at a time (similar to the UNIX more command).

Default value for this variable is false.

enable_recovery_removal_arcs

Controls whether Design Compiler accepts recovery and removal arcs that are specified in the technology library.

Default value for this variable is false.

enable_slew_degradation

Determines whether the transition degradation is taken into account for nets with physical information.

Default value for this variable is true.

enable_special_level_shifter_naming

Setting this variable to true would enable special naming for automatically inserted level shifters

Default value for this variable is false.

estimate_io_latency

Uses estimated I/O latency in timing calculations for ports when true.

Default value for this variable is false.

exit_delete_command_log_file

Controls whether the file specified by the variable command_log_file is deleted after design_analyzer or dc_shell exits normally.

Default value for this variable is false.

exit_delete_filename_log_file

Controls whether the file specified by the variable filename_log_file is deleted after design_analyzer or dc_shell exits normally.

Default value for this variable is true.

filename_log_file

Specifies the name of the filename log file to be used in case a fatal error occurs during execution of design_analyzer or dc_shell.

Default value for this variable is filenames.log.

find_allow_only_non_hier_ports

Controls find command to search for ports in sub-designs (described as hier_ports here).

Default value for this variable is false.

find_converts_name_lists

Controls whether the find command converts the *name_list* string to a list of strings before searching for design objects.

Default value for this variable is false.

find_ignore_case

Controls whether the find command is case-sensitive when matching object names.

Default value for this variable is false.

fsm_auto_inferring

Determines whether or not to automatically extract finite state machine during the compile.

Default value for this variable is false.

fsm_enable_state_minimization

Determines whether or not the state minimization is performed for all finite state machines (FSMs) in the design.

Default value for this variable is false.

fsm_export_formality_state_info

Determines whether or not state machine encoding information is exported into the files that will be used by Formality.

Default value for this variable is false.

fuzzy_matching_enabled

This is a Boolean variable used to enable or disable fuzzy matching.

Default value for this variable is false.

gen_bussing_exact_implicit

Controls whether schematics generated using the create_schematic -implicit command should contain implicit bus names instead of bus rippers.

Default value for this variable is false.

gen_cell_pin_name_separator

Specifies the character used to separate cell names and pin names in the bus names generated by the create_schematic command.

Default value for this variable is /.

gen_create_netlist_busses

Controls whether create_schematic creates netlist buses whenever it creates buses on the schematic.

Default value for this variable is true.

gen_dont_show_single_bit_busses

Controls whether single-bit buses are generated in the schematic.

Default value for this variable is false.

gen_match_ripper_wire_widths

Controls whether the create_schematic command generates rippers whose width always equals the width of the ripped net.

Default value for this variable is false.

gen_max_compound_name_length

Controls the maximum length of compound names of bus bundles (for the create_schematic -sge command).

Default value for this variable is 256.

gen_max_ports_on_symbol_side

Specifies the maximum allowed size of a symbol created by create_schematic.

Default value for this variable is 0.

gen_open_name_postfix

Specifies the postfix to be used by create_schematic -sge when creating placeholder net names for unconnected pins.

Default value for this variable is "".

gen_open_name_prefix

Specifies the prefix to be used by create_schematic -sge when creating placeholder net names for unconnected pins.

Default value for this variable is Open.

gen_show_created_busses

Controls whether a message is printed out every time a schematic bus is created from cell pins for which no equivalent net bus exists in the netlist.

Default value for this variable is false.

gen_show_created_symbols

Controls whether create_schematic prints a warning message every time it generates a new symbol for a cell because an appropriate symbol could not be found in the symbol libraries.

Default value for this variable is false.

gen_single_osc_per_name

Controls whether more than one off-sheet connector with any particular name is drawn on any schematic sheet.

Default value for this variable is false.

generic_symbol_library

Specifies the generic symbol library used for schematics.

Default value for this variable is generic.sdb.

gui_online_browser

Specifies the name of the browser used to invoke the online help system from the help menu of the product

Default value for this variable is netscape.

hdl_keep_licenses

Controls whether HDL licenses that are checked out remain checked out throughout the dc_shell session or are released after use.

Default value for this variable is true.

hdl_preferred_license

Selects an hdl license to check out, if none is currently checked out.

Default value for this variable is "".

hdlin_auto_save_templates

Controls whether HDL designs containing parameters are read in as templates.

Default value for this variable is false.

hdlin_check_input_netlist

Instructs the Verilog netlist reader in dc_shell/psyn_shell to check for the names conflict.

Default value for this variable is false.

hdlin_check_no_latch

Controls whether a warning message is issued if a latch is inferred from a design.

Default value for this variable is false.

hdlin_elab_errors_deep

Default value for this variable is false.

hdlin_enable_assertions

Control Presto HDL compiler's use of SystemVerilog Assertions.

Default value for this variable is false.

hdlin_enable_configurations

Controls the configuration support by Presto VHDL.

Default value for this variable is false.

hdlin_enable_rtldrc_info

Controls whether RTL TestDRC creates file name and line number information for HDL constructs and instances for designs processed by subsequent dc_shell commands.

Default value for this variable is false.

hdlin_ff_always_async_set_reset

Controls whether HDL Compiler checks and reports asynchronous set and reset conditions of flip-flops.

Default value for this variable is true.

hdlin_ff_always_sync_set_reset

Controls whether every constant 0 loaded on a flip-flop under the clock event is used for synchronous reset, and every constant 1 loaded on a flip-flop under the clock event is used for synchronous set.

Default value for this variable is false.

hdlin_field_naming_style

Defines the parts of the net names that Presto HDL Compiler generates corresponding to the fields in VHDL records or in SystemVerilog structs.

Default value for this variable is "".

hdlin_generate_naming_style

Specifies the naming style for *generated* design instances in VHDL designs.

Default value for this variable is %s_%d.

hdlin_generate_separator_style

Specifies the separator string for instances generated in multiple-nested loops.

Default value for this variable is _.

hdlin_ignore_textio_constructs

Controls whether the two commands read and analyze should warn or error when encountering STD.TEXTIO constructs in VHDL code.

Default value for this variable is true.

hdlin_infer_function_local_latches

Controls whether the Presto HDL Compiler infers latches inside functions and tasks.

Default value for this variable is false.

hdlin_infer_multibit

Specifies inference of multibit components for an entire design.

Default value for this variable is default_none.

hdlin_infer_mux

Determines whether and how HDL Compiler infers a MUX_OP.

Default value for this variable is default.

hdlin_keep_signal_name

Determines whether HDL Compiler attempts to keep a signal name.

Default value for this variable is all_driving.

hdlin_latch_always_async_set_reset

Uses, for asynchronous reset, every constant 0 loaded on a latch, and uses, for asynchronous set, every constant 1 loaded on a latch, for a design subsequently analyzed.

Default value for this variable is false.

hdlin_module_arch_name_splitting

Controls whether Presto HDL Compiler recognizes a special format of Verilog module names, which allows users to specify both a module and an implementation architecture.

Default value for this variable is false.

hdlin_module_name_limit

The length threshold for compressing elaborated module names when hdlin_shorten_long_module_name is true.

Default value for this variable is 256.

hdlin_mux_oversize_ratio

Prevents inference of a sparse multiplexer, when the ratio of MUX_OP data inputs to unique data inputs is above the hdlin_mux_oversize_ratio.

Default value for this variable is 100.

hdlin_mux_size_limit

Limits the number of inputs of an inferred multiplexer.

Default value for this variable is 32.

hdlin_mux_size_min

Sets the lower bound for the number of inputs required to infer a multiplexer.

Default value for this variable is 2.

hdlin_mux_size_only

Controls which MUX_OP cells receive the size_only attribute in Presto HDL Compiler.

Default value for this variable is 1.

hdlin_optimize_pla_effort

Controls the effort the Presto HDL Compiler puts into optimization of PLA like constant CASE statements

Default value for this variable is 2.

hdlin_preserve_sequential

Controls whether the elaborate and read commands retain unloaded sequential cells in the design.

Default value for this variable is none.

hdlin_presto_cell_name_prefix

Sets the internal cell name prefix for Presto HDL Compiler.

Default value for this variable is C.

hdlin_presto_net_name_prefix

Sets internal net name prefix for Presto HDL Compiler.

Default value for this variable is N.

hdlin_prohibit_nontri_multiple_drivers

Controls whether the Presto HDL Compiler issues an error, or only a warning, when it finds multiple drivers of a net.

Default value for this variable is true.

hdlin_reporting_level

Determines which information Presto HDL Compiler prints in the report.

Default value for this variable is basic.

hdlin_shorten_long_module_name

Controls whether the Presto HDL Compiler will compress long names of elaborated modules.

Default value for this variable is false.

hdlin_subprogram_default_values

Determines which value the compiler will use as the default value for variables, 'LEFT of its type or 0s.

Default value for this variable is false.

hdlin_sv_ieee_assignment_patterns

Controls the level of support for IEEE-1800 SystemVerilog assignment patterns.

Default value for this variable is 1.

hdlin_sv_packages

Specifies whether and how System Verilog packages should be analyzed.

Default value for this variable is enable.

hdlin_sv_tokens

Specifies whether a tokens file should be written out during the analysis of SystemVerilog designs.

Default value for this variable is false.

hdlin_upcase_names

Controls whether identifiers in the Verilog source code are converted to uppercase letters or left in their original case.

Default value for this variable is false.

hdlin_vhdl93_concat

Controls the concatenation behavior the tool uses to conform to the VHDL '93 Standard or the VHDL '87 Standard.

Default value for this variable is true.

hdlin_vhdl_87

Controls whether VHDL Compiler (Presto) follows VHDL '93 Standard or VHDL '87 Standard.

Default value for this variable is false.

hdlin_vrlg_std

Controls whether Presto Verilog/SystemVerilog enforces Verilog 1995, Verilog 2001, Verilog/SystemVerilog 2005, or SystemVerilog 2009.

Default value for this variable is 2001.

hdlin_while_loop_iterations

Places an upper bound on the number of times a loop is unrolled (to prevent potential infinite loops).

Default value for this variable is 1024.

hdlout_internal_busses

Controls the way in which the write -format verilog command and the write -format vhdl command write out internal bused nets by parsing the names of the nets.

Default value for this variable is false.

hier_dont_trace_ungroup

Disables ungroup tracing set on the design with the ungroup command.

Default value for this variable is 0.

high_fanout_net_pin_capacitance

Specifies the pin capacitance used to compute the loading of high-fanout nets.

Default value for this variable is 1.000000.

high_fanout_net_threshold

Specifies the minimum number of loads for a net to be classified as a high-fanout net.

Default value for this variable is 1000.

hlo_resource_allocation

Sets the default resource sharing type to be used by the compile command, if the resource_allocation attribute is not set.

Default value for this variable is constraint_driven.

ilm_enable_power_calculation

Perform power calculation on design which is to be used as ILM block.

Default value for this variable is true.

ilm_ignore_percentage (not supported in Design Compiler topographical mode)

Specifies a threshold for the percentage of total registers in the transitive fanout of an input port, beyond which the port is to be ignored when identifying interface logic.

Default value for this variable is 25.

ilm_preserve_core_constraints

Enables the ILM mode (ilm_mode), which preserves constraints set on an interface logic model (ILM) core, if set to true.

Default value for this variable is false.

in_gui_session This read-only variable has the value "true" when the GUI is active and the value "false" when the GUI is not active.

Default value for this variable is false.

initial_target_library

Specifies the list of technology libraries of components to be used for the first part of leakage power optimization in place_opt.

Default value for this variable is "".

insert_dft_clean_up

Causes the insert_dft command to use area recovery techniques to reduce the amount of test point logic.

Default value for this variable is true.

insert_test_design_naming_style

Specifies how the insert_dft command names new designs created during the addition of test circuitry.

Default value for this variable is %s_test_%d.

lbo_cells_in_regions

Puts new cells at specific locations within a cluster.

Default value for this variable is false.

level_shifter_naming_prefix

Using this variable, users can specify a prefix for the level shifters names

Default value for this variable is "".

lib_thresholds_per_lib

Causes trip-point values in the Synopsys library to override user-specified values.

Default value for this variable is true.

lib_use_thresholds_per_pin

Causes pin specific trip-point values in the Synopsys library to override Library default trip-point values.

Default value for this variable is true.

libgen_max_differences (not supported in Design Compiler topographical mode)

Specifies to the read_lib command the maximum number of differences to list between the v3.1 format description of a library cell and its statetable description.

Default value for this variable is -1.

link_force_case

Controls the case-sensitive or case-insensitive behavior of the link command.

Default value for this variable is check_reference.

link_library

Specifies the list of design files and libraries used during linking.

Default value for this variable is * your_library.db .

ltl_obstruction_type

Controls the routing blockage type for the named obstructions, without route type being specified.

Default value for this variable is placement_only.

mcmm_high_capacity_effort_level

Controls the behavior of Multi-corner/Multi_mode (MCMM) scenario reduction.

Default value for this variable is 0.

monitor_cpu_memory

Displays the CPU time, elapsed time, and peak memory usage before and after each core command.

Default value for this variable is false.

mux_auto_inferring_effort

Specifies the MUX inferring effort level.

Default value for this variable is 2.

mv_allow_ls_on_leaf_pin_boundary

Sets the variable to true to allow level-shifter insertion on leaf pin (such as macro cell pin) boundaries.

Default value for this variable is false.

mw_cell_name

Contains the Milkyway design cell name.

Default value for this variable is "".

mw_design_library

Contains the Milkyway design library.

Default value for this variable is "".

mw_disable_escape_char

Specifies to disable the escape characters for hierarchy delimiter.

Default value for this variable is true.

mw_hdl_bus_dir_for_undef_cell

Specify how to determine the bus direction for undefined cell.

Default value for this variable is 0.

mw_hdl_expand_cell_with_no_instance

This variable determines whether to expand netlist cells without instances or not.

Default value for this variable is false.

mw_hvo_core_filler_cells

Generate core filler cell instances for HVO.

Default value for this variable is true.

mw_hvo_corner_pad_cells

Generate corner pad cell instances for HVO.

Default value for this variable is true.

mw_hvo_diode_ports

Generate diode ports for HVO.

Default value for this variable is false.

mw_hvo_dump_master_names

Specify the cell instance masters that must be dumped out for HVO.

Default value for this variable is "".

mw_hvo_empty_cell_definition

Generate definition of empty cells for HVO.

Default value for this variable is false.

mw_hvo_generate_macro_definition

Generate macro declarations for HVO.

Default value for this variable is false.

mw_hvo_output_onezero_for_pg

Generate 1'b1 for power and 1'b0 for ground nets for HVO.

Default value for this variable is true.

mw_hvo_output_wire_declaration

Generate wire declarations of nets for HVO.

Default value for this variable is false.

mw_hvo_pad_filler_cells

Generate pad filler cell instances for HVO.

Default value for this variable is true.

mw_hvo_pg_nets

Generate power and ground nets for HVO.

Default value for this variable is true.

mw_hvo_pg_ports

Generate power and ground ports for HVO.

Default value for this variable is false.

mw_hvo_split_bus

Generate individual bus bits for HVO.

Default value for this variable is false.

mw_hvo_strip_backslash_before_hiersep

Do not generate backslashes before hierarchy seperators for HVO.

Default value for this variable is true.

mw_hvo_unconnected_cells

Generate unconnected cell instances for HVO.

Default value for this variable is true.

mw_hvo_unconnected_ports

Generate unconnected cell ports for HVO.

Default value for this variable is false.

mw_logic0_net

Contains the equivalent logic0 net for the design.

Default value for this variable is VSS.

mw_logic1_net

Contains the equivalent logic1 net for the design.

Default value for this variable is VDD.

mw_reference_library

Contains the Milkyway reference libraries.

Default value for this variable is "".

optimize_reg_always_insert_sequential

Controls whether the optimize_registers command will remove and reinsert the sequential elements in the circuits even if no register was moved.

Default value for this variable is false.

optimize_reg_max_time_borrow

Specifies the maximum amount of time borrowing at all latches when retiming latches using the optimize_registers command. A negative value means there is no limit on borrowing other than the one resulting from the clock period.

Default value for this variable is -1048576.0.

optimize_reg_retime_clock_gating_latches

Specifies whether to move clock gating latches during retiming of latches.

Default value for this variable is false.

pdefout_diff_original

Used in conjunction with the option *-new_cells_only* of the command write_clusters.

Default value for this variable is true.

physopt_area_critical_range (Design Compiler topographical mode only)

Specifies a margin of slack for cells during area optimization. If a cell has a slack less than the area critical range, area optimization is not done for the cell.

Default value for this variable is -1.04858e+06.

physopt_cpu_limit (Design Compiler topographical mode only)

This variable is obsolete and cannot be enabled. Please use set_physopt_cpulimit_options command instead.

Default value for this variable is 0.

physopt_create_missing_physical_libcells (Design Compiler topographical mode only)

Directs Physical Compiler to create dummy physical descriptions of missing physical library cells.

Default value for this variable is false.

physopt_enable_extractor_rc (Design Compiler topographical mode only)

Enables and disables the support of extractor-based RC computation.

Default value for this variable is true.

physopt_enable_router_process (Design Compiler topographical mode only)

Enables and disables the use of a separate process to perform routing.

Default value for this variable is true.

physopt_enable_tlu_plus (Design Compiler topographical mode only)

Enables and disables the support of TLU+ based RC computation.

Default value for this variable is true.

physopt_enable_tlu_plus_process (Design Compiler topographical mode only)

Enables and disables using a separate process to perform TLU+ based RC extraction.

Default value for this variable is true.

physopt_enable_via_res_support (Design Compiler topographical mode only)

Enables and disables the support of via resistance for virtual route RC estimation.

Default value for this variable is false.

physopt_hard_keepout_distance (Design Compiler topographical mode only)

Specifies the keepout distance used by the physopt, create_placement, and legalize_placement commands.

Default value for this variable is 0.

physopt_ignore_lpin_fanout

The tool ignore max fanout constraints which are specfied in the library.

Default value for this variable is false.

physopt_power_critical_range (Design Compiler topographical mode only)

Specifies a margin of slack for cells during leakage power optimization. If a cell has a slack less than the power critical range, power optimization will not be done for the cell.

Default value for this variable is -1.04858e+06.

placer_enable_enhanced_router (Design Compiler topographical mode only)

Enables a mode of coarse placement in which congestion removal is done with the global router.

Default value for this variable is false.

placer_max_cell_density_threshold (Design Compiler topographical mode only)

Enables a mode of coarse placement in which cells can clump together.

Default value for this variable is -1.

placer_run_in_separate_process (Design Compiler topographical mode only)

Enables and disables the use of a separate process to perform placement.

Default value for this variable is true.

placer_soft_keepout_channel_width (Design Compiler topographical mode only)

Specifies a soft keepout distance that is used by the place_opt and create_placement commands.

Default value for this variable is 0.

plot_box

Causes a box to be drawn around the plot. The default is *false*.

Default value for this variable is false.

plot_command

Specifies the operating system command that produces a hard copy of the plot.

Default value for this variable is lpr -Plw.

plot_orientation

Specifies whether the schematic is vertical or horizontal.

Default value for this variable is best_fit.

plot_scale_factor

Specifies a scaling factor for the schematic.

Default value for this variable is 100.

plotter_maxx

Specifies the x coordinate of the upper right corner of the plot output device.

Default value for this variable is 584.

plotter_maxy

Specifies the y coordinate of the upper right corner of the plot output device.

Default value for this variable is 764.

plotter_minx

Specifies the x coordinate of the lower left corner of the plot output device.

Default value for this variable is 28.

plotter_miny

Specifies the y coordinate of the lower left corner of the plot output device.

Default value for this variable is 28.

port_complement_naming_style

Defines the convention the compile command uses to rename ports complemented as a result of using the set_boundary_optimization command.

Default value for this variable is %s_BAR.

power_cg_all_registers

Specifies to the insert_clock_gating command whether to clock gate all registers, including those that do not meet the necessary requirements.

Default value for this variable is false.

power_cg_auto_identify

Activates automatic identification of Power Compiler inserted clock gating circuitry from a structural netlist.

Default value for this variable is false.

power_cg_balance_stages

Controls clock gate stage balancing is on or off during compile [-incremental_mapping] -gate_clock or compile_ultra [-incremental_mapping] -gate_clock.

Default value for this variable is false.

power_cg_cell_naming_style

Specifies the naming style for clock gating cells created during insert_clock_gating.

Default value for this variable is "".

power_cg_derive_related_clock

When *true*, clock domain relationship between registers will be derived from the hierarchical context.

Default value for this variable is false.

power_cg_designware

Performs clock gating on DesignWare sequential components in the design.

The use of power_cg_designware variable will be obsolete in a future release. Clock gating insertion with compile_ultra -gate_clock automatically inserts clock gates in DesignWare modules.

Default value for this variable is false.

power_cg_enable_alternative_algorithm

Specifies to the insert_clock_gating, compile-gate_clock and compile_ultra -gate_clock commands whether to use an alternative algorithm to find gatable registers.

Default value for this variable is false.

power_cg_flatten

Specifies to different ungroup commands whether to flatten Synopsys clock-gating cells.

Default value for this variable is false.

power_cg_gated_clock_net_naming_style

Specifies the naming style for gated clock nets created during insert_clock_gating.

Default value for this variable is "".

power_cg_ignore_setup_condition

When *true*, the setup condition will be ignored for latch-free clock gating.

Default value for this variable is false.

power_cg_inherit_timing_exceptions

Specifies that during compile -gate_clock or compile_ultra [-incr] -gate_clock, timing exceptions defined on registers have to be automatically inferred on to the enable pin of the clock gate that is gating these registers.

Default value for this variable is false.

power_cg_module_naming_style

Specifies the naming style for clock gating modules created during insert_clock_gating.

Default value for this variable is "".

power_cg_print_enable_conditions

When *true*, the enable conditions of registers and clock gates will be reported during clock gate insertion.

Default value for this variable is false.

power_cg_print_enable_conditions_max_terms
Specifies the maximum number of product terms to be reported in the sum of product expansion of the enable condition.

Default value for this variable is 10.

power_cg_reconfig_stages
Controls the reconfiguration of multistage clock gates during compile [-incremental_mapping] -gate_clock or compile_ultra [-incremental_mapping] -gate_clock.

Default value for this variable is false.

power_default_static_probability
Specifies the default static probability value.

Default value for this variable is 0.5.

power_default_toggle_rate
Specifies the default toggle rate value.

Default value for this variable is 0.1.

power_default_toggle_rate_type
Specifies the default toggle rate type.

Default value for this variable is fastest_clock.

power_do_not_size_icg_cells
Controls whether compile does not size the integrated clock-gating cells in a design to correct DRC violations because doing so may result in lower area and power.

Default value for this variable is true.

power_driven_clock_gating

Controls whether switching activity and dynamic power of the register banks should be considered when optimizing the clock gating of the design.

Default value for this variable is false.

power_enable_one_pass_power_gating

When *true*, one-pass flow power gating will be enabled.

Default value for this variable is false.

power_enable_power_gating

When set to true compile will enable the power gating flow which allows the selected retention registers from target library to be used to map sequential elements.

Default value for this variable is false.

power_fix_sdpd_annotation

Specifies whether user-annotated SDPD switching activity annotation is corrected before it is used.

Default value for this variable is true.

power_fix_sdpd_annotation_verbose

Specifies whether verbose messages are reported during fixing of user-annotated SDPD switching activity.

Default value for this variable is false.

power_hdlc_do_not_split_cg_cells

When *true*, nsert_clock_gating does not split clock-gating cells to limit their fanout.

Default value for this variable is false.

power_keep_license_after_power_commands

Affects the amount of time a Power Compiler license is checked out during a dc_shell (Design Compiler) session.

Default value for this variable is false.

power_lib2saif_rise_fall_pd

Specifies whether lib2saif generates forward SAIF files with directives to generate rise/fall dependent path-dependent toggle counts.

Default value for this variable is false.

power_min_internal_power_threshold

Specifies the minimum cell internal power value that can be used in power calculations.

Default value for this variable is "".

power_model_preference

Specifies the preference between the CCS power and the NLPM models in library cells that have power specified in both models.

Default value for this variable is nlpm.

power_opto_extra_high_dynamic_power_effort

This variable makes the compile command invoke more dyanmic power optimization algorithms.

Default value for this variable is false.

power_preserve_rtl_hier_names

Preserves the hierarchy information of the RTL objects in the RTL design.

Default value for this variable is false.

power_rclock_inputs_use_clocks_fanout

Specifies whether clock network objects in an input port fanout are used to infer the input port's related clock.

Default value for this variable is true.

power_rclock_unrelated_use_fastest

Specifies whether the fastest clock is set as the related clock of a design object when a related clock is not inferred by the related clock inference mechanism.

Default value for this variable is true.

power_rclock_use_asynch_inputs

Specifies whether the inferred related clock on an asynchronous pin of a flip-flop is used to determine the nferred related clock on the cell's outputs.

Default value for this variable is false.

power_remove_redundant_clock_gates

Specifies to the compile -incremental and physopt -incremental commands whether to remove redundant Synopsys clock gating cells.

Default value for this variable is true.

power_rtl_saif_file

Defines for the rtl2saif command where to store the forward-annotation SAIF file, if you do not specify the -output option.

Default value for this variable is power_rtl.saif.

power_sa_propagation_effort

Specifies the default effort level used when propagating switching activity.

Default value for this variable is low.

power_sa_propagation_verbose

Specifies the default verbose mode used when propagating switching activity.

Default value for this variable is false.

power_same_switching_activity_on_connected_objects

Forces the tool to use the last user-annotated switching activity data on all connected tool objects.

Default value for this variable is false.

power_sdpd_message_tolerance

Specifies the tolerance value for issuing warnings and information messages during fixing of user-annotated SDPD switching activity.

Default value for this variable is 0.00001.

power_sdpd_saif_file

Defines for the lib2saif command where to store the forward-annotation SAIF file, if you do not specify the -output option.

Default value for this variable is power_sdpd.saif.

psyn_stress_map (Design Compiler topographical mode only)

Enables generation of stress map. Set this variable to true to generate stress maps from Physical Compiler.

Default value for this variable is false.

rc_degrade_min_slew_when_rd_less_than_rnet

Enables or disables the use of slew degradation in min analysis mode during the RCCALC-009 condition.

Default value for this variable is false.

rc_driver_model_mode

Specifies which driver model type to use for RC delay calculation.

Default value for this variable is basic.

rc_input_threshold_pct_fall

Specifies the threshold voltage that defines the startpoint of the falling cell or net delay calculation.

Default value for this variable is 50.000000.

rc_input_threshold_pct_rise

Specifies the threshold voltage that defines the startpoint of the rising cell or net delay calculation.

Default value for this variable is 50.000000.

rc_noise_model_mode

When set to advanced, enables the use of CCS noise, if available in the design library.

Default value for this variable is basic.

rc_output_threshold_pct_fall

Specifies the threshold voltage that defines the startpoint of the falling cell or net delay calculation.

Default value for this variable is 50.000000.

rc_output_threshold_pct_rise

Specifies the threshold voltage that defines the startpoint of the rising cell or net delay calculation.

Default value for this variable is 50.000000.

rc_receiver_model_mode

Specifies which receiver model type to use for RC delay calculation.

Default value for this variable is basic.

rc_slew_derate_from_library

Specifies the derating needed for the transition times in the Synopsys library to match the transition times between the characterization trip points.

Default value for this variable is 1.000000.

rc_slew_lower_threshold_pct_fall

Specifies the threshold voltage that defines the endpoint of the falling slew calculation.

Default value for this variable is 20.000000.

rc_slew_lower_threshold_pct_rise

Specifies the threshold voltage that defines the startpoint of the rising slew calculation.

Default value for this variable is 20.000000.

rc_slew_upper_threshold_pct_fall

Specifies the threshold voltage that defines the startpoint of the falling slew calculation.

Default value for this variable is 80.000000.

rc_slew_upper_threshold_pct_rise

Specifies the threshold voltage that defines the endpoint of the rising slew calculation.

Default value for this variable is 80.000000.

read_db_lib_warnings (not supported in Design Compiler topographical mode)

Indicates that warnings are to be printed while a technology .db library is being read in with the read command. When false (the default), no warnings are given.

Default value for this variable is false.

read_translate_msff

Indicates (when *true*, the default) that master-slave flip-flops (specified with the clocked_on_also syntax) are to be automatically translated to master-slave latches. When *false*, both master and slave remain flip-flops.

Default value for this variable is true.

register_duplicate

Controls whether compile should invoke Register duplication or not.

Default value for this variable is false.

reoptimize_design_changed_list_file_name (not supported in Design Compiler topographical mode)

Creates a file in which to store the list of cells that changed and cells and nets that were added during post-layout or in-place optimization.

Default value for this variable is "".

report_default_significant_digits

Sets the default number of significant digits for many reports.

Default value for this variable is -1.

rom_auto_inferring

Inferring ROM from RTL description.

Default value for this variable is true.

rp_shift_column_for_fixed_cells (Design Compiler topographical mode only)

Determines whether the tool allows the horizontal shifting of relative placement columns in the presence of single tap cells.

Default value for this variable is false.

rtl_load_resistance_factor

Specifies a factor to be used by the set_rtl_load command to calculate resistance values from capacitance values for RTL loads.

Default value for this variable is 0.0.

sdc_write_unambiguous_names

Ensures that cell, net, pin, lib_cell, and lib_pin names that are written to the SDC file are not ambiguous.

Default value for this variable is true.

sdfout_allow_non_positive_constraints (not supported in Design Compiler topographical mode)

Writes out PATHCONSTRAINT constructs with nonpositive (<= 0) constraint values. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is false.

sdfout_min_fall_cell_delay (not supported in Design Compiler topographical mode)

Specifies the minimum non-back-annotated fall cell delay that the write_timing command writes to a timing file in SDF format. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is 0.000000.

sdfout_min_fall_net_delay (not supported in Design Compiler topographical mode)

Specifies the minimum non-back-annotated fall net delay that write_timing can write to a timing file in SDF format. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is 0.000000.

sdfout_min_rise_cell_delay (not supported in Design Compiler topographical mode)

Specifies the minimum non-back-annotated rise cell delay that write_timing can write to a timing file in SDF format. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is 0.000000.

sdfout_min_rise_net_delay (not supported in Design Compiler topographical mode)

Specifies the minimum non-back-annotated rise net delay that the write_timing command can write to a timing file in SDF format. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is 0.000000.

sdfout_time_scale (not supported in Design Compiler topographical mode)

Specifies the time scale of the delays written to timing files in SDF format. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is 1.000000.

sdfout_top_instance_name (not supported in Design Compiler topographical mode)

Specifies the name prepended to all instance names when writing timing files in SDF format. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is "".

sdfout_write_to_output (not supported in Design Compiler topographical mode)

Specifies whether the write_timing -f sdf command writes interconnect delays between cells and top-level output ports. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is false.

search_path

Specifies directories that the tool searches for files specified without directory names.

Default value for this variable is {fIsearch_pathfP + .}.

sh_allow_tcl_with_set_app_var

Allow set_app_var and get_app_var commands to work with application variables

Default value for this variable is Application specific.

sh_allow_tcl_with_set_app_var_no_message_list

Suppress CMD-104 messages for variables in this list

Default value for this variable is Application specific.

sh_arch

Indicates the system architecture of your machine.

Default value for this variable is Platform-dependent.

sh_command_abbrev_mode

Sets the command abbreviation mode for interactive convenience.

Default value for this variable is Application specific.

sh_command_log_file

Specifies the name of the file to which is written a log of the initial values of variables and executed commands.

Default value for this variable is command.log.

sh_continue_on_error

Allows processing to continue when errors occur during script execution with the source command.

Default value for this variable is Application specific.

sh_dev_null

Indicates the current null device.

Default value for this variable is Platform-dependent.

sh_enable_line_editing

Enables the command line editing capabilities. This variable is for use in Tcl mode only.

Default value for this variable is true.

sh_enable_page_mode

Displays long reports one page at a time (similar to the UNIX more command.

Default value for this variable is Application specific.

sh_enable_stdout_redirect

Allow the redirect command to capture output to the Tcl stdout channel.

Default value for this variable is Application specific.

sh_line_editing_mode

Enables vi or emacs editing mode. This variable is for use in Tcl mode only.

Default value for this variable is emacs.

sh_new_variable_message

Controls a debugging feature for tracing the creation of new variables.

Default value for this variable is Application specific.

sh_new_variable_message_in_proc

Controls a debugging feature for tracing the creation of new variables in a Tcl procedure.

Default value for this variable is false.

sh_new_variable_message_in_script

Controls a debugging feature for tracing the creation of new variables within a sourced script.

Default value for this variable is false.

sh_output_log_file This read-write variable is used to name the file which all console output is captured. It can be set to an empty string to disable output capture.

Default value for this variable is "".

sh_product_version

Indicates the version of the application currently running.

Default value for this variable is .

sh_script_stop_severity

Indicates the error message severity level which would cause a script to stop executing before it completes.

Default value for this variable is Application specific.

sh_source_emits_line_numbers

Indicates the error message severity level which would cause an informational message to be issued listing the script name and line number where that message occurred.

Default value for this variable is Application specific.

sh_source_logging

Indicates if individual commands from a sourced script should be logged to the command log file.

Default value for this variable is Application specific.

sh_source_uses_search_path

Causes the search command to use the search_path variable to search for files. This variable is for use in dc_shell-t (Tcl mode of dc_shell) only.

Default value for this variable is true.

sh_tcllib_app_dirname

Indicates the name of a directory where application-specific Tcl files are found.

Default value for this variable is .

sh_user_man_path

Indicates a directory root where the user can store man pages for display with the man command.

Default value for this variable is The empty list.

si_use_partial_grounding_for_min_analysis

Affects the behavior of report_timing and compile with crosstalk effect is enabled. .SH

Default value for this variable is false.

si_xtalk_reselect_delta_and_slack

Reselect nets that satisfy both delta delay and slack reselection criteria.

Default value for this variable is false.

si_xtalk_reselect_delta_delay

Specifies the threshold of net delay change caused by crosstalk analysis, above which IC Compiler reselects the net for subsequent delay calculations.

Default value for this variable is 5.

si_xtalk_reselect_delta_delay_ratio

Specifies the threshold of the ratio of net delay change caused by crosstalk analysis to the total stage delay, above which IC-Compiler reselects a net for subsequent delay calculations.

Default value for this variable is 0.95.

si_xtalk_reselect_max_mode_slack

Specifies the max mode pin slack threshold, below which IC-Compiler reselects a net for subsequent delay calculations.

Default value for this variable is 0.

si_xtalk_reselect_min_mode_slack

Specifies the min mode pin slack threshold, below which IC-Compiler reselects a net for subsequent delay calculations.

Default value for this variable is 0.

single_group_per_sheet

Specifies to the tool to put only one logic group on a sheet.

Default value for this variable is false.

site_info_file

Contains the path to the site information file for licensing.

Default value for this variable is "".

sort_outputs

Sorts output ports on the schematic by port name.

Default value for this variable is false.

suppress_errors

Specifies a list of error codes for which messages are to be suppressed during the current Design Analyzer/dc_shell session.

Default value for this variable is PWR-18 OPT-932 OPT-317 RCCALC-010 RCCALC-011.

symbol_library

Specifies the symbol libraries to use during schematic generation.

Default value for this variable is your_library.sdb.

synlib_abort_wo_dw_license

Abort compile command if DesignWare license is required to compile the current design, but the DesignWare license is not available.

Default value for this variable is false.

synlib_dont_get_license

Specifies a list of synthetic library part licenses that the compiler does not automatically check out.

Default value for this variable is "".

synlib_dwgen_smart_generation

Determines strategies used by the DesignWare arithmetic generators.

Default value for this variable is true.

synlib_hiis_force_on_cells

Specifies a list of design cells on which the compiler is to force hierarchical incremental implementation selection (hiis).

Default value for this variable is "".

synlib_iis_use_netlist

Allows netlists of the DesignWare parts to be used, instead of timing models, for cost comparison during the Incremental Implementation Selection step.

Default value for this variable is false.

synlib_wait_for_design_license

Specifies a list of authorized synthetic library licenses that Design Compiler is to wait for.

Default value for this variable is "".

synopsys_program_name

Indicates the name of the program currently running.

Default value for this variable is .

synopsys_root

Indicates the root directory from which the application was run.

Default value for this variable is .

syntax_check_status

Reports whether the syntax_check mode is enabled.

Default value for this variable is false.

synthetic_library

Specifies a list of synthetic libraries to use when compiling.

Default value for this variable is "".

systemcout_debug_mode

Default value for this variable is false.

systemcout_levelize

Levelizes and flattens the netlist and replaces standard DesignWare operations with simulatable SystemC, before writing out the netlist, during write -f systemc command activity.

Default value for this variable is true.

target_library

Specifies the list of technology libraries of components to be used when compiling a design.

Default value for this variable is your_library.db.

template_naming_style

Generates automatically a unique name when a module is built.

Default value for this variable is %s_%p.

template_parameter_style

Generates automatically a unique name when a module is built.

Default value for this variable is %s%d.

template_separator_style

Generates automatically a unique name when a module is built.

Default value for this variable is _.

test_allow_clock_reconvergence

Allows reconvergent nets to originate from the same top-level clock port.

Default value for this variable is true.

test_bsd_allow_tolerable_violations

Allows the optimize_bsd command to replace observe_and_control BSR cells with observe_only cells or remove BSR cells during timing-driven or area-driven optimization.

Default value for this variable is false.

test_bsd_control_cell_drive_limit

Specifies the number of cells a single BSR control cell can drive while optimizing control cell allocation during optimize_bsd command activity

Default value for this variable is 0.

test_bsd_default_bidir_delay

Defines the default switching time of bidirectional ports in a tester cycle for bsd applications.

Default value for this variable is 0.0.

test_bsd_default_delay

Defines the default time in a tester cycle to apply values to input ports for bsd applications.

Default value for this variable is 0.0.

test_bsd_default_strobe

Defines the default strobe time in a test cycle for output ports and bidirectional ports in output mode for bsd applications.

Default value for this variable is 95.0.

test_bsd_default_strobe_width

Defines the default strobe pulse width, which is the default time that specifies how long after invocation the strobe pulse needs to be held active for bsd applications.

Default value for this variable is 0.0.

test_bsd_manufacturer_id

Specifies the manufacturer ID to use to create the value captured in the device identification register during execution of the insert_bsd command.

Default value for this variable is 0.

test_bsd_optimize_control_cell

When true, allows the optimize_bsd command to optimize allocation of BSR control cells during area-driven optimization, using the value of the test_bsd_control_cell_drive_limit variable.

Default value for this variable is false.

test_bsd_part_number

Specifies the part number to use to create the value captured in the device identification register during execution of the insert_bsd command.

Default value for this variable is 0.

test_bsd_version_number

Specifies the version number to use to create the value captured in the device identification register during execution of the insert_bsd command.

Default value for this variable is 0.

test_bsdl_default_suffix_name

Specifies the default suffix for the name of the BSDL file generated by the write_bsdl command.

Default value for this variable is bsdl.

test_bsdl_max_line_length

Specifies the maximum number of characters per line for the output BSDL file the write_bsdl command produces.

Default value for this variable is 80.

test_capture_clock_skew

Specifies a qualitative measure of clock skew.

Default value for this variable is small_skew.

test_cc_ir_masked_bits

Identifies instruction register (IR) bits to be masked during the search by the check_bsd command for all possible implemented instructions.

Default value for this variable is 0.

test_cc_ir_value_of_masked_bits

Specifies values to be forced into bits of the instruction register (IR) that are masked, during the search by the check_bsd command for all possible implemented instructions.

Default value for this variable is 0.

test_check_port_changes_in_capture

Checks (through the check_test command) for changes in values applied to bidirectional ports in the parallel measure cycle.

Default value for this variable is true.

test_clock_port_naming_style

Specifies the naming style used by the insert_scan command for global test signal ports created in designs during the addition of test circuitry.

Default value for this variable is test_c%s.

test_dedicated_subdesign_scan_outs

Instructs DFT Compiler to create dedicated scan-out ports on subdesigns.

Default value for this variable is false.

test_default_bidir_delay

Defines the default switching time of bidirectional ports in a tester cycle.

Default value for this variable is 0.0.

test_default_delay

Defines the default time in a tester cycle to apply values to input ports.

Default value for this variable is 0.0.

test_default_period

Defines the default length of a test vector cycle.

Default value for this variable is 100.0.

test_default_scan_style

Defines the default scan style for the insert_dft command if a scan style is not specified with the set_scan_style command.

Default value for this variable is multiplexed_flip_flop.

test_default_strobe

Defines the default strobe time in a test cycle for output ports and bidirectional ports in output mode.

Default value for this variable is 40.0.

test_default_strobe_width

Defines the default strobe pulse width, which is the default time that specifies how long after invocation the strobe pulse needs to be held active.

Default value for this variable is 0.0.

test_design_analyzer_uses_insert_scan (not supported in Design Compiler topographical mode)

Executes (when *true*) the insert_scan command through a Design Analyzer menu.

Default value for this variable is true.

test_disable_find_best_scan_out

Selects the scan-out pin on a scan cell based on availability instead of timing slack.

Default value for this variable is false.

test_dont_fix_constraint_violations

Minimizes performance constraint violations.

Default value for this variable is false.

test_enable_capture_checks

Controls checking for capture violations during execution of the check_dft command.

Default value for this variable is true.

test_infer_slave_clock_pulse_after_capture

Guides protocol inference for master/slave test design methodologies during execution of the check_test command.

Default value for this variable is infer.

test_isolate_hier_scan_out

Prevents the insert_dft command inserting logic that isolates scan connections at hierarchical boundaries during functional operation.

Default value for this variable is 0.

test_jump_over_bufs_invs

Determines whether or not insert_scan and preview_scan consider output pins of buffers and inverters to be internal clocks.

Default value for this variable is true.

test_mode_port_inverted_naming_style

Specifies the naming style to use for the test_hold_logic_zero type of test mode signal ports to be created in the design.

Default value for this variable is test_mode_i%s.

test_mode_port_naming_style

Determines the naming style to be used by insert_dft command for test mode ports created in designs during the addition of test point circuitry.

Default value for this variable is test_mode%s.

test_mux_constant_si

Specifies how scan insertion uses a port you declare as scan input, when the port is tied high or to the ground in functional mode.

Default value for this variable is false.

test_mux_constant_so

Specifies how scan insertion uses a port you declare as scan output, when the port is tied high or to the ground in functional mode.

Default value for this variable is false.

test_non_scan_clock_port_naming_style

Specifies the style the insert_dft command uses to name the ports that clock gating creates for nonscan clocks.

Default value for this variable is test_nsc_%s.

test_point_keep_hierarchy

Synthesizes (when *true*) test points and ungroups the test point design, during execution of the insert_dft command.

Default value for this variable is false.

test_preview_scan_shows_cell_types

Shows (when *true*) cell instance types, during execution of the preview_scan command.

Default value for this variable is false.

test_protocol_add_cycle

Adds an extra cycle (when *true*) after the shift cycle, in the test protocol, during execution of the check_test command.

Default value for this variable is true.

test_rtldrc_latch_check_style

Specifies the latch check style to use during rtldrc command activities.

Default value for this variable is default.

test_scan_clock_a_port_naming_style

Determines the naming style to be used by the insert_scan command for test scan clock a ports created in designs during the addition of test scan circuitry.

Default value for this variable is test_sca%s.

test_scan_clock_b_port_naming_style

Determines the naming style to be used by the insert_scan command for test scan clock b ports created in designs during the addition of test scan circuitry.

Default value for this variable is test_scb%s.

test_scan_clock_port_naming_style

Determines the naming style used by the insert_scan command for global test scan signal ports created in designs during the addition of test circuitry.

Default value for this variable is test_sc%s.

test_scan_enable_inverted_port_naming_style

Determines the naming style to be used by the insert_scan command for scan enable inverted ports created in designs during the addition of test scan circuitry.

Default value for this variable is test_sei%s.

test_scan_enable_port_naming_style

Determines the naming style to be used by the insert_scan command for test scan enable ports created in designs during the addition of test scan circuitry.

Default value for this variable is test_se%s.

test_scan_in_port_naming_style

Specifies the naming style used by the insert_scan command for serial test-signal ports created in designs during the addition of test circuitry.

Default value for this variable is test_si%s%s.

test_scan_link_so_lockup_key

Indicates to the preview_scan command what key to use to identify cells with scan-out lock-up latches in reports.

Default value for this variable is l.

test_scan_link_wire_key

Indicates to the preview_scan command the key to use to identify cells that drive wire-scan links in reports.

Default value for this variable is w.

test_scan_out_port_naming_style

Used the same as test_scan_in_port_naming_style .

Default value for this variable is test_so%s%s.

test_scan_segment_key

Tells the preview_scan command what key to use to identify scan segments in reports.

Default value for this variable is s.

test_scan_true_key

Specifies to the preview_scan command the key to use to identify in reports cells with true scan attributes.

Default value for this variable is t.

test_setup_additional_clock_pulse

When this variable is set to true, an extra clock cycle is added to all clocks in the design in the initialization procedure during execution of the create_test_protocol command.

Default value for this variable is false.

test_simulation_library

Indicates the pathname to a Verilog simulation library.

Default value for this variable is .

test_stil_max_line_length

Specifies the maximum line length for the file written by the write_test_protocol -format stil command.

Default value for this variable is 72.

test_stil_multiclock_capture_procedures

Indicates to the write_test_protocol -format stil command to create capture procedures in the STIL protocol, with multiple clocks active in each procedure.

Default value for this variable is false.

test_stil_netlist_format

Indicates to the write_test_protocol command what netlist format to use when writing out STIL protocol files.

Default value for this variable is db.

test_use_test_models

Indicates to DFT Compiler to create or use test models when performing scan insertion.

Default value for this variable is false.

test_user_defined_instruction_naming_style

Indicates to the check_bsd command and the write_bsdl command the naming style to use for the user-defined (nonstandard) instructions inferred by these commands.

Default value for this variable is USER%d.

test_user_test_data_register_naming_style

Indicates to the check_bsd command and the write_bsdl command the naming style to use for the user-defined (nonstandard) test data registers inferred by these commands.

Default value for this variable is UTDR%d.

test_write_four_cycle_stil_protocol

Instructs the write_test_protocol -format stil command to insert in the output STIL protocol file a dummy cycle between all measure and capture cycles in the STIL protocol.

Default value for this variable is false.

text_editor_command (not supported in Design Compiler topographical mode)

Specifies the command that executes when the Edit/File menu is selected in the Design Analyzer text window.

Default value for this variable is xterm.

text_print_command (not supported in Design Compiler topographical mode)

Specifies the command that executes when the File/Print menu is selected in the Design Analyzer text window.

Default value for this variable is lpr.

timing_check_defaults

define the default check list in check_timing command.

Default value for this variable is generated_clock loops no_input_delay unconstrained_endpoints pulse_clock_cell_type no_driving_cell partial_input_delay.

timing_clock_gating_propagate_enable

Allow the gating enable signal delay to propagate through the gating cell.

Default value for this variable is false.

timing_crpr_remove_clock_to_data_crp

Allows the removal of Clock Reconvergence Pessimism (CRP) from paths that fan out directly from clock source to the data pins of sequential devices.

Default value for this variable is false.

timing_crpr_threshold_ps

Specifies amount of pessimism that clock reconvergence pessimism removal (CRPR) is allowed to leave in the report.

Default value for this variable is 20.

timing_disable_cond_default_arcs

Disable the default, non-conditional timing arc between pins that do have conditional arcs.

Default value for this variable is false.

timing_edge_specific_source_latency

Controls whether the generated clock source latency computation will consider edge relationship or not.

Default value for this variable is false.

timing_enable_multiple_clocks_per_reg

Enables or disables analysis of multiple clocks that reach a single register.

Default value for this variable is false.

timing_enable_non_sequential_checks

Enables or disables library non_sequential checks in the design.

Default value for this variable is false.

timing_gclock_source_network_num_master_registers

The maximum number of register clock pins clocked by the master clock allowed in generated clock source latency paths.

Default value for this variable is 1.

timing_input_port_clock_shift_one_cycle (not supported in Design Compiler topographical mode)

Determines whether or not paths originating at input ports are given an extra cycle to meet their timing constraints. This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is true.

timing_input_port_default_clock

Determines whether a default clock is assumed at input ports for which you have not defined a clock-specific input external delay.

Default value for this variable is true.

timing_remove_clock_reconvergence_pessimism

Enables or disables clock reconvergence pessimism removal.

Default value for this variable is false.

timing_report_attributes (not supported in Design Compiler topographical mode)

Specifies the list of attributes to be reported with the report_timing -attributes command.

Note: This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is {dont_touch dont_use map_only size_only ideal_net}.

timing_self_loops_no_skew

Affects the behavior, runtime, and CPU usage of report_timing and compile.

Note: This variable will be obsolete in the next release. Please adjust your scripts accordingly.

Default value for this variable is false.

timing_separate_clock_gating_group

Specifies if a separate cost group is used for clock gating checks in timing analysis, reports, and optimization.

Default value for this variable is false.

timing_use_clock_specific_transition

Propagate the transition from the specific clock path for latency calculation

Default value for this variable is true.

timing_use_driver_arc_transition_at_clock_source

Uses the backward cell arc to compute a realistic driver model at the driver pin for primary clock sources and also generated clock that can not trace back to its master clock

Default value for this variable is true.

timing_use_enhanced_capacitance_modeling

Specifies the use of the following attributes found on a library pin: rise_capacitance_range(low, high), fall_capacitance_range(low, high), rise_capacitance, and fall_capacitance.

Default value for this variable is false.

ungroup_keep_original_design

Controls ungroup and compile command to keep the original design when a design is ungrouped in XG mode.

Default value for this variable is false.

uniquify_keep_original_design

Controls uniquify command to keep the original design when a multiply instantiated design is uniqufied in XG mode.

Default value for this variable is false.

uniquify_naming_style

Specifies the naming convention to be used by the uniquify command.

Default value for this variable is %s_%d.

upf_extension

Sets the variable to false to disable writing of UPF extension commands in save_upf.

Default value for this variable is true.

use_port_name_for_oscs

Specifies that when off-sheet connectors for nets also have ports on them, they are given the name of the port.

Default value for this variable is false.

verbose_messages

Causes more explicit system messages to be displayed during the current Design Analyzer dc_shell session.

Default value for this variable is true.

verilogout_equation

Writes Verilog "assign" statements (Boolean equations) for combinational gates, rather than gate instantiations.

Default value for this variable is false.

verilogout_higher_designs_first

Writes Verilog "modules" so that the higher level designs come before lower level designs, as defined by the design hierarchy.

Default value for this variable is false.

verilogout_ignore_case

Instructs the compiler not to consider case when comparing identifiers to Verilog reserved words.

Default value for this variable is false.

verilogout_include_files

Specifies to the write -f verilog command to write an include statement that will have the name of the value you set for this variable.

Default value for this variable is "".

verilogout_no_tri

Declares three-state nets as Verilog "wire" instead of "tri." This variable is useful in eliminating "assign" primitives and "tran" gates in the Verilog output.

Default value for this variable is false.

verilogout_show_unconnected_pins

Instructs the Verilog writer in dc_shell to write out all of the unconnected instance pins, when connecting module ports by name. For example, modb b1 (.A(in),.Q(out),.Qn()).

Default value for this variable is false.

verilogout_single_bit

Instructs the compiler not to output vectored ports in the Verilog output. All vectors are written as single bits.

Default value for this variable is false.

verilogout_unconnected_prefix

Instructs the Verilog writer in dc_shell to use the name SYNOPSYS_UNCONNECTED_ to create unconnected wire names. The general form of the name is SYNOPSYS_UNCONNECTED_%d.

Default value for this variable is SYNOPSYS_UNCONNECTED_.

vhdllib_architecture (not supported in Design Compiler topographical mode)

Determines the VHDL model types for the write_lib command to generate.

Default value for this variable is VITAL.

vhdllib_glitch_handle (not supported in Design Compiler topographical mode)

Specifies whether timing hazards are to have glitch-forced (glitch on detect) or spike-forced (glitch on event) Xs.

Default value for this variable is true.

vhdllib_logic_system (not supported in Design Compiler topographical mode)

Specifies the logic system in which the tool is to create the VHDL libraries.

Default value for this variable is ieee-1164.

vhdllib_logical_name (not supported in Design Compiler topographical mode)

This variable defines the logical name to be used by the VHDL libraries. If the variable is set to an empty string (the default), the file base name is used as the default.

Default value for this variable is "".

vhdllib_negative_constraint (not supported in Design Compiler topographical mode)

Determines whether a generated VITAL model is to have negative constraint handling capability.

Default value for this variable is false.

vhdllib_sdf_edge

Determines whether edge information is to be added to delay generics in VITAL libraries.

Default value for this variable is false.

vhdllib_tb_compare (not supported in Design Compiler topographical mode)

Controls library testbench generation. No testbenches are created if this variable is set to 0 (the default).

Default value for this variable is 0.

vhdllib_tb_x_eq_dontcare (not supported in Design Compiler topographical mode)

Specifies the default value for the testbenches X_Eq_DontCare generic Boolean flag. If X_Eq_DontCare is true, X states are ignored during output comparisons.

Default value for this variable is false.

vhdllib_timing_checks (not supported in Design Compiler topographical mode)

Determines the default value of the cell TimingChecksOn generic Boolean flag in the VITAL model.

Default value for this variable is true.

vhdllib_timing_mesg (not supported in Design Compiler topographical mode)

Determines the value of the GlitchMode parameter of the VitalPropagatePathDelay procedure in the VITAL model.

Default value for this variable is true.

vhdllib_timing_xgen (not supported in Design Compiler topographical mode)

Determines the default value of the XGenerationOn generic Boolean flag in the VITAL model.

Default value for this variable is false.

vhdllib_vital_99

Determines whether VITAL libraries are to incorporate changes for VITAL 99.

Default value for this variable is false.

vhdlout_bit_type

Sets the basic bit type in a design written to VHDL.

Default value for this variable is std_logic.

vhdlout_bit_vector_type

Sets the basic bit vector type in a design written to VHDL.

Default value for this variable is std_logic_vector.

vhdlout_dont_create_dummy_nets

Instructs the VHDL writer not to create dummy nets to connect unused pins or ports in your design.

Default value for this variable is false.

vhdlout_equations

Defines how the tool is to write combinational logic and sequential logic.

Default value for this variable is false.

vhdlout_follow_vector_direction

Specifies how the tool is to use the original range direction when it writes out an array.

Default value for this variable is true.

vhdlout_lower_design_vector

Determines the way in which the write -f vhdl command writes out ports on lower-level designs

Default value for this variable is true.

vhdlout_one_name

Determines the literal name for constant bit value 1 in a design written in VHDL.

Default value for this variable is 1'.'

vhdlout_package_naming_style

Determines the name the tool is use for the type conversion packages written out by the VHDL writer (VHDLout).

Default value for this variable is CONV_PACK_%d.

vhdlout_preserve_hierarchical_types

Affects the way in which the write -f vhdl command writes out ports on lower-level designs

Default value for this variable is VECTOR.

vhdlout_separate_scan_in

Affects the way in which the write -f vhdl command writes out the scan chain in VHDL.

Default value for this variable is false.

vhdlout_single_bit

Affects the way in which the write -f vhdl command writes out ports on the top-level design.

Default value for this variable is USER.

vhdlout_target_simulator

Names the target simulator to which the the tool writes the VHDL file. Valid value is xp.

Default value for this variable is "".

vhdlout_three_state_name

Names the high-impedance bit value used for three-state device values.

Default value for this variable is Z'.'

vhdlout_three_state_res_func

Names a user-supplied three-state resolution function that must be in one of the packages specified by the vhdlout_use_packages variable.

Default value for this variable is "".

vhdlout_top_configuration_arch_name

Determines the name of the outside architecture, depending on the value you defined for the vhdlout_write_top_configuration variable, and causes the VHDL writer (VHDLout) to write out a configuration statement.

Default value for this variable is A.

vhdlout_top_configuration_entity_name

Determines the name of the outside entity, depending on the value you defined for the vhdlout_write_top_configuration variable, and causes the VHDL writer (VHDLout) to write out a configuration statement.

Default value for this variable is E.

vhdlout_top_configuration_name

Determines the name of the configuration statement the write -f vhdl command writes out, when the vhdlout_write_top_configuration variable is set to true.

Default value for this variable is CFG_TB_E.

vhdlout_top_design_vector

Affects the way in which the write -f vhdl command writes out ports on the top-level design.

Default value for this variable is false.

vhdlout_unconnected_pin_prefix

Affects the way in which the write -f vhdl command writes out unconnected pin names.

Default value for this variable is n.

vhdlout_unknown_name

Specifies the value the tool is to use to drive a signal to the "unknown" state.

Default value for this variable is X'.'

vhdlout_upcase

Default value for this variable is .

vhdlout_use_packages

Instructs the write -f vhdl command to write into the VHDL file a clause called a "use clause," which contains a list of package names for each of the packages described in this man page, for all entities.

Default value for this variable is IEEE.std_logic_1164.

vhdlout_wired_and_res_func

Specifies the name of a "wired and" resolution function.

Default value for this variable is "".

vhdlout_wired_or_res_func

Specifies the name of a "wired or" resolution function.

Default value for this variable is "".

vhdlout_write_architecture

Instructs the write -format vhdl command to write out architecture declarations.

Default value for this variable is true.

vhdlout_write_attributes

This variable is obsolete.

Default value for this variable is .

vhdlout_write_components

Instructs the write -format vhdl command to write out component declarations for cells mapped to a technology library.

Default value for this variable is true.

vhdlout_write_entity

Instructs the write -format vhdl command to write out entity declarations.

Default value for this variable is true.

vhdlout_write_top_configuration

Instructs the write -format vhdl command to write out a configuration statement, if necessary, such as when ports on the top-level design are written as vectors instead of user types.

Default value for this variable is false.

vhdlout_zero_name

Determines the literal name for constant bit value 0 in a design written in VHDL.

Default value for this variable is 0'.'

view_analyze_file_suffix (not supported in Design Compiler topographical mode)

Specifies, in a list of file extensions, the files shown in the File/Analyze dialog box of Design Analyzer.

Default value for this variable is {v vhd vhdl}.

view_arch_types

Sets the contents of the architecture option menu. Contains a list of host machine architectures you can use for background jobs from the Design Analyzer viewer.

Default value for this variable is sparcOS5 hpux10 rs6000 sgimips.

view_background

Specifies the background color of the Design Analyzer viewer.

Default value for this variable is black.

view_cache_images (not supported in Design Compiler topographical mode)

Specifies to Design Analyzer that the tool is to cache bitmaps for fast schematic drawing.

Default value for this variable is true.

view_command_log_file

Names a file and its location that is to contain all text written to the Design Analyzer Command window.

Default value for this variable is "./view_command.log".

view_command_win_max_lines (not supported in Design Compiler topographical mode)

Contains the maximum number of lines to be saved in the Design Analyzer command window.

Default value for this variable is 1000.

view_dialogs_modal (not supported in Design Compiler topographical mode)

Requires that the question and error dialogs in Design Analyzer be confirmed, before you can continue entering commands.

Default value for this variable is true.

view_disable_cursor_warping (not supported in Design Compiler topographical mode)

Causes the cursor to be automatically "warped" (moved). When *false*, the cursor is automatically "warped" (or moved) to dialog boxes.

Default value for this variable is true.

view_disable_error_windows (not supported in Design Compiler topographical mode)

Instructs Design Analyzer not to post the error windows when errors occur.

Default value for this variable is false.

view_disable_output (not supported in Design Compiler topographical mode)

Disables output to the Design Analyzer command window.

Default value for this variable is false.

view_error_window_count (not supported in Design Compiler topographical mode)

Specifies the maximum number of errors Design Analyzer reports for a command.

Default value for this variable is 6.

view_execute_script_suffix

Displays only files with the stated suffixes, from directories you select in the Execute Script option window of the Setup menu of Design Analyzer.

Default value for this variable is ".script .scr .dcs .dcv .dc .con .tcl".

view_info_search_cmd (not supported in Design Compiler topographical mode)

Invokes, if set, the online information viewer through the optional menu item On-Line Information.

Default value for this variable is "".

view_log_file (not supported in Design Compiler topographical mode)

Specifies the file in which the tool stores events that occur in the viewer.

Default value for this variable is "".

view_on_line_doc_cmd (not supported in Design Compiler topographical mode)

Invokes, if set, the online documentation viewer, through the optional menu item On-Line Documentation.

Default value for this variable is "".

view_read_file_suffix

Displays only files with the stated suffixes, from directories you select with the Read option of the File menu of Design Analyzer.

Default value for this variable is db gdb sdb edif eqn fnc lsi mif NET pla st tdl v vhd vhdl xnf.

view_report_append

Specifies to the tool to append to the specified file the reports the Design Vision menus generate.

Default value for this variable is true.

view_report_interactive

Specifies to the tool to send to the command line view the reports generated by Design Vision menus.

Default value for this variable is true.

view_report_output2file

Specifies to the tool to send to the specified file the reports generated by Design Vision menus.

Default value for this variable is false.

view_script_submenu_items (not supported in Design Compiler topographical mode)

Allows users to add to the Design Analyzer Setup pulldown menu valid items to invoke user scripts.

Default value for this variable is {}.

view_set_selecting_color (not supported in Design Compiler topographical mode)

Specifies the color to use for selecting and zooming.

Default value for this variable is .

view_tools_menu_items (not supported in Design Compiler topographical mode)

Permits partial configuration of the Tools pulldown menu to add a new menu item for invoking user scripts.

Default value for this variable is {}.

view_use_small_cursor (not supported in Design Compiler topographical mode)

Specifies to the tool that the X display is to support only 16 x 16-bit map size cursors.

Default value for this variable is "".

view_use_x_routines (not supported in Design Compiler topographical mode)

Enables the use of internal arc-drawing routines (instead of X routines).

Default value for this variable is true.

view_write_file_suffix

Displays only files with the stated suffixes, from directories you select with the Save As option of the File menu of Design Analyzer.

Default value for this variable is gdb db sdb do edif eqn fnc lsi NET neted pla st tdl v vhd vhdl xnf.

write_name_nets_same_as_ports

Specifies to the tool that nets are to receive the same names as the ports the nets are connected to.

Default value for this variable is false.

write_sdc_output_lumped_net_capacitance

Determines whether or not the write_sdc command outputs net loads.

Default value for this variable is true.

write_sdc_output_net_resistance

Determines whether or not the write_sdc command outputs net resistance.

Default value for this variable is true.

write_test_formats

Specifies the test vector formats recognized and created by the write_test command.

Default value for this variable is synopsys tssi_ascii tds verilog vhdl wgl.

write_test_include_scan_cell_info

Specifies to the write_test command to include in the vector files scan-chain, cell, and inversion information for vector formats.

Default value for this variable is true.

write_test_input_dont_care_value

Controls the logic value the write_test command outputs when you have an input with a don't care condition.

Default value for this variable is X.

write_test_max_cycles

Controls the automatic partitioning of long test sets across multiple files, by specifying the maximum number of tester cycles any one vector file can contain.

Default value for this variable is 0.

write_test_max_scan_patterns

Controls the automatic partitioning of long test sets across multiple files, by specifying the maximum number of scan test patterns any one vector file can contain.

Default value for this variable is 0.

write_test_new_translation_engine

Specifies to the write_test command to choose to use new translation engine or old engine to do test program translation.

Default value for this variable is false.

write_test_pattern_set_naming_style

Specifies how to name pattern sets when long test sets are partitioned across multiple files.

Default value for this variable is TC_Syn_%d.

write_test_round_timing_values

Specifies to the write_test command to round to the nearest integer all timing values.

Default value for this variable is true.

write_test_scan_check_file_naming_style

Specifies how to name the file containing the vectors that test the scan chain logic.

Default value for this variable is %s_schk.%s.

write_test_vector_file_naming_style

Specifies how to name scan vector files, when long test sets are partitioned across multiple files.

Default value for this variable is %s_%d.%s.

write_test_vhdlout

Determines whether the write_test -format vhdl command generates a VHDL test program in TEXTIO format.

Default value for this variable is inline.

x11_set_cursor_background

Specifies background color of the cursor in the Design Analyzer menus and viewer.

Default value for this variable is "".

x11_set_cursor_foreground

Specifies foreground color of the cursor in the Design Analyzer menus and viewer.

Default value for this variable is magenta.

x11_set_cursor_number

Specifies the cursor, from the standard X cursor font used by the Design Analyzer menus and viewer.

Default value for this variable is -1.

xterm_executable (not supported in Design Compiler topographical mode)

Specifies the path to an xterm program spawned to run Synopsys analysis tools (for example, RTL Analyzer or BCView). The default is *xterm*.

Default value for this variable is xterm.